

Daffodil International University

Department of Software Engineering SE-431

Project/thesis Project Documentation

**Enhancing Personalized Purchase Decisions by Integrating Buyer Psychology: A Dual-Role
E-Commerce Platform Designed for Seamless Interaction**

Submitted by

Chowdhury Md.

Imtiazul Islam

ID: 201-35-564

Department of Software Engineering

Daffodil International University

Supervised by

MD Rajib Mia

Lecturer

Department of Software Engineering

Daffodil International University

This Project report has been submitted in fulfilment of the requirements for the Degree

of Bachelor of Science in Software Engineering.

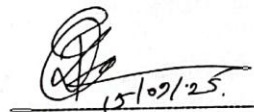
DECLARATION

I hereby declare that I have done this project under the supervisor of **Md.Rajib Mia, Lecturer(Senior Scale)**,Department of Software Engineering, Daffodil International University. I also declare that this project is my original work for the degree of B.Sc. in Software Engineering and neither the whole work nor any part of this project has been submitted for another degree in this or any other university.



Md.Mahfuzur Rahman
ID: 152-35-1249
Department of Software Engineering,
Daffodil International University

Certified by:



Md.Rajib Mia
Lecturer (Senior Scale)
Department of Software Engineering,
Daffodil International University

APPROVAL

This project titled on "Enhancing Personalized Purchase Decisions by Integrating Buyer Psychology: A Dual-Role E-Commerce Platform Designed for Seamless Interaction" submitted by Chowdhury Md. Imtiazul Islam (ID: 201-35-564) to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

BOARD OF EXAMINERS



Dr. Imran Mahmud
Professor & Head
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Chairman



Md Shohel Arman
Assistant Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 1



Md. Rajib Mia
Lecturer (Senior Scale)
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 2



Md Habibur Rahman
Associate Professor
Department of Computer Science and Engineering
Islamic University, Bangladesh

External Examiner

ACKNOWLEDGEMENT

I'm Chowdhury Md. Imtiazul Islam. Rajib Mia, a Senior Scale lecturer in the Department of Software Engineering (SWE) and my supervisor, was a huge help during the planning and development stages of the system. Without his smart advice and strategic guidance, I would not have been able to reach the final phase of my development. By developing the original idea and providing ongoing support, he played a significant role in the project's success.

I am truly thankful to my family and friends for always supporting me.

ABSTRACT

This project offers a creative and intuitive e-commerce application to improve the online shopping experience by making buying and selling easier for everyone. By enabling users the choice to join as either people who buy or sell, the system promotes smooth communication within a single platform. One of the most important parts of the application is the Product Details Screen, which stays visible even when users switch between tabs. This design is based on user behavior and increases the likelihood that customers will buy a product they have previously enjoyed because they prefer to return to it if they don't find something better. Consumers can search for products, browse through product details, and filter by category without erasing their browsing history. In addition, they may change their profile, track orders, store favorite items, and manage their basket. Sellers can handle orders, upload and manage products, and view helpful business analytics on a separate stage. In addition to concentrating on functionality, this app uses basic psychology concepts to enhance user interaction. As a result, both buyers and sellers enjoy a more considerate and interesting purchasing experience.

TABLE OF CONTENT

INDEX

Approval	i
Declaration	ii
Acknowledgement	iii
Abstract	iv
Chapter 1 Introduction	1-2
1.1 Project Overview	1
1.2 Project Purpose	1
1.2.1 Proposed System	2
Chapter 2 System Analysis	3-8
2.1 Feasibility Analysis	3
2.1.1 Technical Feasibility	3
2.1.2 Operational Feasibility	3-4
2.2 Functional Requirements	4-6
2.3 System Requirements	7
2.4 Non-Functional Requirements	7
2.5 Performance	8
Chapter 3 System Design	9-42
3.1 System Development Approach	9
3.1.1 Functional Diagram	10
3.1.2 Functional Diagram Descriptions	11-17
3.2 Activity Diagram	
3.3	
3.3.1 Workflow Diagram: Buyer	18
3.2. Workflow Diagram: Buyer	19

3.2.3 Workflow Diagram: Buyer	20
3.2.4 Workflow Diagram: Buyer	21
3.2.5 Workflow Diagram: Buyer	22
3.2.6 Workflow Diagram: Buyer	23
3.2.7 Workflow Diagram: Seller	24
3.2.8 Workflow Diagram: Seller	25
3.2.9 Workflow Diagram: Seller	26
3.2.10 Workflow Diagram: Seller	27
3.2.11 Workflow Diagram: Admin	28
3.3 Interaction Diagram	29-34
3.3.1 Interaction Diagram: Buyer	29
3.3.2 Interaction Diagram: Seller	30
3.3.3 Interaction Diagram: Seller	31
3.3.4 Interaction Diagram: Admin	32
3.3.5 Interaction Diagram: Admin	33
3.3.6 Interaction Diagram: Admin	34
3.4 ER Model Diagram	35
3.5 Class Diagram	36-39
3.5.1 Class Model: Buyer	36
3.5.2 Class Model: Seller	37
3.5.3 Class Model: Seller	38
3.5.4 Class Model: Admin	39
3.6 Block Diagram	39-41
3.6.1 System Architecture Model: Buyer	39
3.6.2 System Architecture Model: Seller	40
3.6.3 System Architecture Model: Admin	41
Chapter 4 Development Tool and Technology	42

4.1 Integrated Development Environment (IDE)	42
4.2 Programming Language	42
4.3 User Interface Design	42
4.4 Database	42
4.5 Deployment and Hosting	42
Chapter 5 Application Testing	43-52
5.1 Test Functions	44
5.1.1 Feature to check	44-46
5.2 Testing Strategies	47
5.2.1 Test Approach	47-48
5.2.2 Testing Schedule	49
5.3 Test Case	50-52
5.3.1 User Authentication & Access Control	50
5.3.2 Functionality	51
5.3.3 Usability	52
Chapter 6 User Manual	53-62
6.1 Landing Page	53-62
Chapter 7 Conclusion	63-66
7.1 Project Link	63
7.2 Limitations	64
7.3 Future Scope	65
Reference	66

CHAPTER 1: INTRODUCTION

1.1 Project Overview

To provide buyers & sellers with a smooth, personalized shopping experience, this project showcases a mobile e-commerce application developed using the Flutter framework. This app is unique because it focuses on consumer behavior, particularly how consumers go back to products they enjoyed when they can't find better alternatives. The product detail screen remains accessible even when users navigate the app in order to facilitate this.

Customers can quickly manage their cart and orders, search, filter, and explore products, and save favorites. Through their own dashboard, sellers can upload products, monitor orders, and see basic sales data.

Focusing on user needs, practical design, and intelligent interaction features that mirror real-world usage and behavior, this The project has undergone meticulously designed to meet the final defense requirements.

1.2 Project Purpose

This e-commerce application's primary objective is to better understand the thoughts and actions of consumers in order to make online shopping more intuitive and personalized. The purpose of the app is to facilitate easy communication between buyers and sellers on a single platform.

Among the project's main goals are:

Smarter Purchase Experience:

Because customers frequently return to products they have expressed interest in, it is thought that keeping the product detail screen easily accessible will help them make better purchasing decisions.

Dual-Role Platform:

This dual-role platform facilitates both buyers and sellers with features like browsing, filtering, cart management, and seller-side product and order control.

Behavior-Based Design:

Include buyer psychology in the design to boost the possibility of a purchase and encourage repeat visits to the product.

Cross-Platform Support:

This feature ensures wide accessibility and was created with Flutter to run smoothly on iOS and Android smartphones.

This project aims to increase the effectiveness and engagement of the shopping experience while also giving sellers a simple way to manage their businesses. Its emphasis on both usability and user behavior makes it a potent response to the demands of modern e-commerce.

1.2.1 Proposed System

The proposed e-commerce system aims to improve the online shopping experience by making it more personalized, interactive, and easy to use. It serves both buyers and sellers and prioritizes user-friendly navigation and behavior-driven design to help users make better decisions.

Some of the main characteristics of the suggested system are as follows:

- **Dual-Role User Access**
Buyers can track orders, sellers can manage their products, and both parties have easily navigable dashboards and separate modules.
- **Persistent Product View**
It automatically keeps product detail pages open or accessible because it assumes that customers regularly review products before making a purchase.
- **Personalized Experience**
Utilizes buyer behavior patterns, like managing favorites, filtering by preferences, and going back to prior interests, to direct the user journey.
- **Smooth User Interface (UI)**
Flutter was used to create a quick, easy, and clean interface that works well on both iOS and Android.
- **Scalable and Secure Architecture**
Its design made use of Firebase support and modular code to offer storage, authentication, and real-time data.

Offering a seamless and intelligent e-commerce platform where buyers and sellers can communicate effectively and benefit from an intuitive, behavior-based design is the primary objective of the proposed system.

Chapter 2: SYSTEM ANALYSIS

2.1 Feasibility Analysis

A feasibility study evaluates the proposed dual-role e-commerce platform's usefulness, viability, and practicality. This analysis considers both the technical and operational aspects of the system.

2.1.1 Technical Feasibility

The system was developed using the Flutter framework, which makes cross-platform development for iOS and Android easy. The technical implementation is dependable and effective thanks to integration with Firebase, which provides scalable backend services, secure authentication, and live database support.

2.1.2 Operational Feasibility

The app is designed with both Buyers and Sellers in mind, offering clear navigation, easy onboarding, and personalized features. These user-friendly elements make the platform easy to adopt and use, improving overall engagement and satisfaction for both user roles.

2.2 Functional Requirements

- Buyers can register and log in using their email and password.
- Access is blocked for invalid credentials; login requires email and password validation.
- Users can choose to remember their login credentials for faster access.
- Buyers will be able to browse products, search by name or category, and view product details.
- Customers can choose items to add to their cart, complete transactions, save favorites, and check the

status of their orders (completed or active).

- Customers have the ability to modify their login credentials and manage their profile information.
- The program will allow password reset via OTP sent to the registered email.
- Buyers will be able to access the app as a guest with limited features.
- Sellers are able to create an account and safely access it by logging in.
- Product entries can be added, edited, and removed by sellers.
- Sellers can view incoming orders, update order status (processing, shipped, completed), and manage stock.
- Merchants are allowed to view sales history and product performance analytics.
- The system will offer a shared home screen with personalized views for both buyers and sellers.
- Admins will have the ability to approve or reject seller registrations and manage flagged content.
- Admins can manage categories, view user activity, and maintain platform guidelines.
- All users can access support, FAQs, About Us, Terms & Conditions, and Privacy Policy from the settings.

2.3 System Requirements

➤ **Hardware Requirements:**

- Servers:
 - Multi-core processors for handling multiple user requests and background operations smoothly.
 - Minimum 16 GB RAM for optimal performance, especially during peak traffic.
 - At least 500 GB SSD storage to store user data, product images, orders, and transaction history.
 - Reliable internet connectivity and backup power to ensure continuous uptime.

- End-User Devices:
 - Compatible with smartphones utilizing Android 8.0 (Oreo) or later.
 - iOS devices running Apple iOS 10 or higher.
 - Minimum 2 GB RAM for mobile devices recommended for smooth app performance.

➤ **Database Requirements:**

- Product & Order Data:
 - Product information (name, category, description, price, stock status, image URL).
 - Order records including buyer ID, seller ID, order date, payment status, and delivery status.
- User Profiles:
 - Buyer and Seller accounts with details like name, email, password (hashed), phone number, address, etc.
 - Admin account credentials and permission levels.
- Transaction & Activity Logs:
 - Records of login attempts, failed logins, order updates, and other critical actions.

➤ **Security Requirements:**

- User Authentication:
 - Secure login system with hashed and salted passwords using a strong algorithm (e.g., SHA-256 or crypt).
 - OTP verification for account recovery and sensitive operations (like password reset).
- Authorization & Roles:
 - RBAC is used by the platform to manage and distinguish between admin, seller, and buyer user permissions.
- Data Protection:
 - End-to-end encryption for sensitive data transmission (e.g., using HTTPS with SSL/TLS).
 - Secure local storage using device-level encryption for user credentials (if remembered).
 - Regular backups and logging for recovery and auditing purposes.

2.4 Non-Functional Requirements

- **Operational Speed:**
The program must launch quickly and respond to user inputs immediately with minimal delay, ensuring smooth browsing, searching, and checkout experiences.
- **User Interaction Quality:**
Both buyers and sellers should find the application's interface easy to use, straightforward, and navigable.
- **Scalability:**
The system should support growing numbers of users, products, and transactions without performance degradation.
- **Security:**
Secure login, Data that is safely stored, and Access control by user role must be implemented to protect user information and transactions.
- **Ability to access:**
There should be very little downtime and high system availability.
- **Maintainability:**
The codebase and backend system should be modular and easy to update or debug for future improvements or bug fixes.
- **Reliability:**
All core features, such as account creation, order placement, and payment processing, must function consistently without failure.
- **Cross-Platform Support:**
The application must work flawlessly on a variety of iOS and Android smartphones, supporting common OS versions and screen sizes.

2.5 Performance

- Response Time:
 - **Goal:** Ensure smooth and quick interactions throughout the app.
 - **Benchmark:** Most user actions (like adding to cart, placing an order, or browsing products) should respond within **1–2 seconds**.
- Page Load Speed:
 - **Goal:** Load product listings, images, and user dashboards without delay.
 - **Benchmark:** Aim for page/component load times under **3 seconds**, even with multiple products or categories.
- Error Handling:
 - **Goal:** Handle unexpected issues gracefully with clear and friendly error messages.
 - **Benchmark:** Maintain low error rates and minimize app crashes or transaction failures.
- User Experience (UX):
 - **Goal:** Deliver a seamless shopping experience across both buyer and seller journeys.

Benchmark: Achieve high user retention and satisfaction scores through fast, reliable performance

Chapter 3: System Design

3.1 System Design Approach:

Waterfall Model used in developing this e-commerce application, as the project requirements are clearly defined and stable throughout the development cycle.

Why Waterfall:

1. Offers a **step-by-step structure** that's easy to manage.
2. Ensures **complete documentation** at each stage.
3. Ideal for projects with **low risk and fixed scope**.
4. Helps maintain **tight control** over timelines and deliverables.

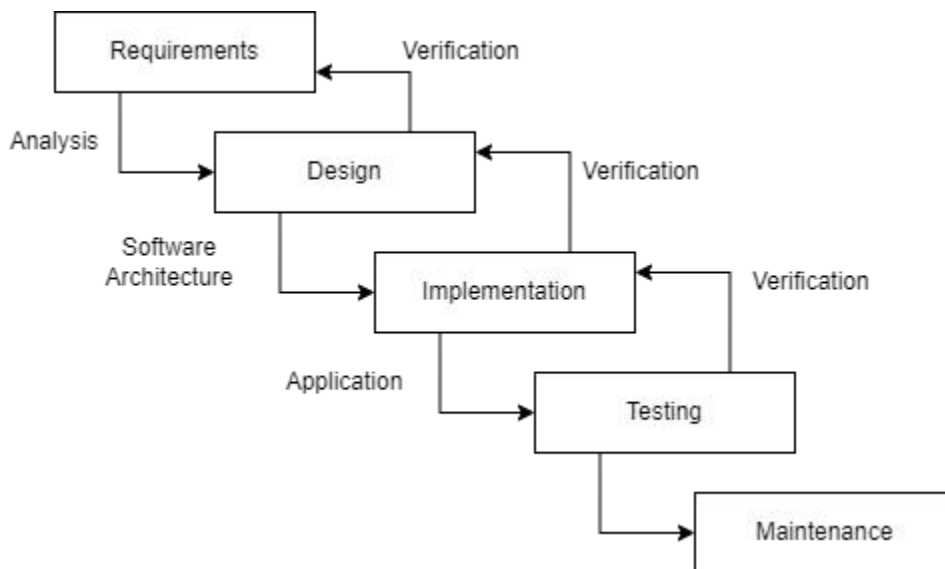


Figure: Waterfall model

3.1.2 Functional Diagram Descriptions

1.

User scenario Name	Buyer Login
Actor	Buyer
Pre-Condition	<ol style="list-style-type: none">1. Application shows a list of search results for the user that match their query.
Flow and Counts	<ol style="list-style-type: none">1. Buyer opens the app and submits login information.2. For authentication, application verifies the user login information.3. Buyer is directed to the home screen following a successful verification.
Alternate Path	<ol style="list-style-type: none">1. If the login details are wrong, the system issues a warning message.2. In case if account doesn't exist, prompts to sign up.

2.

Use case Name	Browse and Search Products
Actor	Buyer
Pre-Condition	1 Buyer is logged in or accessing as a guest.
Flow and Counts	<ol style="list-style-type: none">1. Buyer enters a search term or selects a category.2. System fetches and displays matching products.3. Buyer can view product details.
Alternative Flow	If no products match, system shows "No results found."

3.

Use case Name	Add to Cart
Actor	Buyer
Pre-Condition	1. Buyer is logged in.
Flow and Counts	1. The buyer selects a product and adds it to the shopping cart. 2. System confirms that item is placed in the Purchase cart.
Alternative Scenario	1. In case product is already in the cart, system may show a warning or increase quantity.

4.

User scenario Name	Place Purchase
Role	Buyer
Initial Condition	<ol style="list-style-type: none">1. The user's cart contains selected products.
Flow and Counts	<ol style="list-style-type: none">1. Buyer reviews the cart and proceeds to checkout.2. Selects payment and delivery options.3. System confirms order and updates order status..
Alternative Scenario	<ol style="list-style-type: none">1. When products are out of stock, the system displays a notification and stops the checkout.

5.

Use case Name	View Order History
Actor	Buyer
Pre-Condition	<ol style="list-style-type: none">1. Buyer is logged in.
Flow and Counts	<ol style="list-style-type: none">1. Buyer navigates to "Orders" section.2. System displays ongoing and completed orders.3. Buyer can tap to view order details.
Alternative Flow	<ol style="list-style-type: none">1. If no orders exist, show message "No orders placed yet."

6.

Use case Name	Upload Product
Actor	Seller
Pre-Condition	<ol style="list-style-type: none">1. Seller is logged in and verified.
Flow and Counts	<ol style="list-style-type: none">1. Seller taps "Add Product."2. Enters details like name, price, category, image.3. Submits product.4. System verifies and adds to catalog.
Alternative Flow	<ol style="list-style-type: none">1. When required fields are not filled, the system shows an error message and stops the submission process.

7.

User scenario Name	User Account Administration
Role	Administrator
Initial Condition	1. System Manager is authenticated.
Flow and Counts	<ol style="list-style-type: none"> 1. Admin views all registered users (Buyers and Sellers). 2. Can edit, delete, or suspend accounts.
Alternative Flow	<ol style="list-style-type: none"> 1. If an action is attempted on invalid user, the system shows an error.

8.

User scenario Name	Admin Manage Inventory Items
Role	System Manager
Pre-Condition	1. System Manager is authenticated.
Flow and Counts	1. Admin can view and manage all products on the platform. 2. Can delete, approve, or flag inappropriate content.
Alternative Flow	1. If product doesn't exist, show error message.

3.2 Workflow Diagram:

3.2.1 Workflow Diagram: Buyer

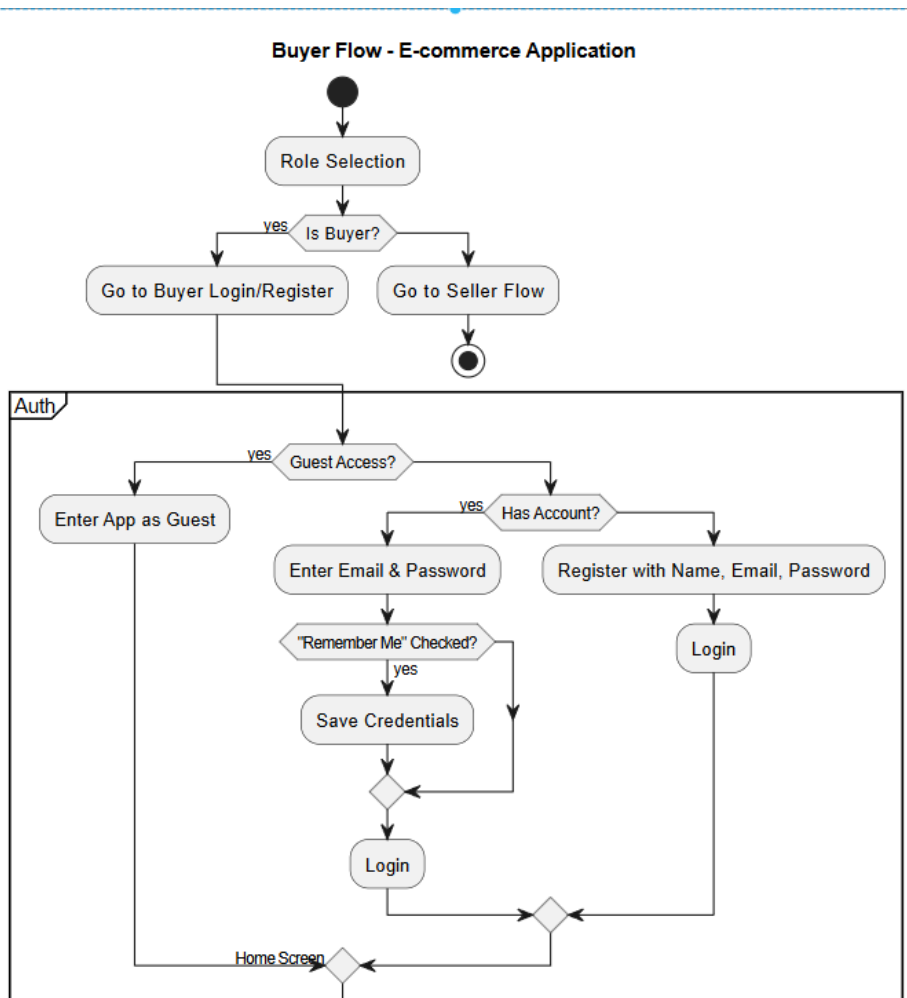


Figure: Workflow Diagram 1.1

3.2.2 Activity Diagram: Bayer

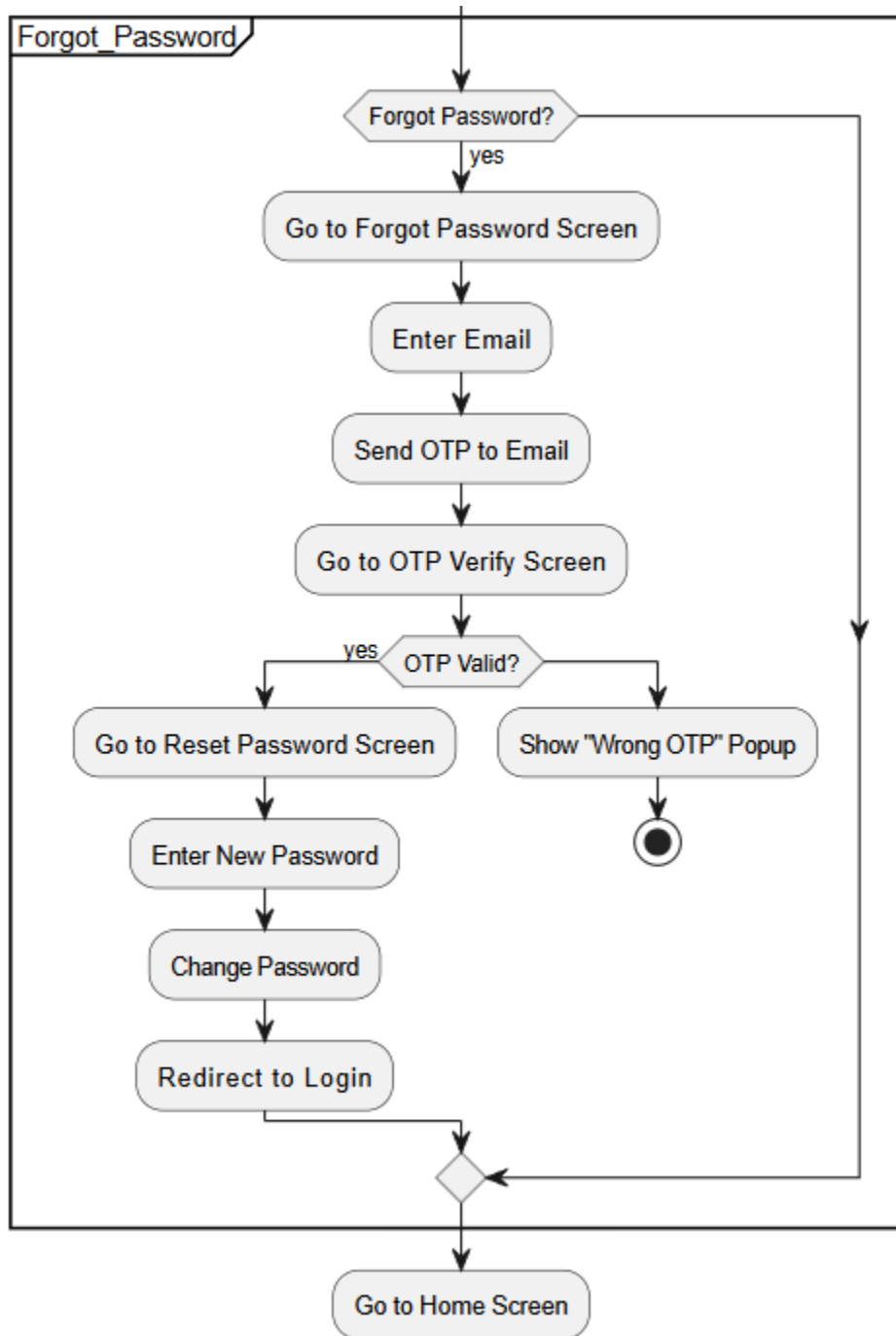


Figure: Workflow Diagram 1.2

3.2.3 Workflow Diagram: Bayer

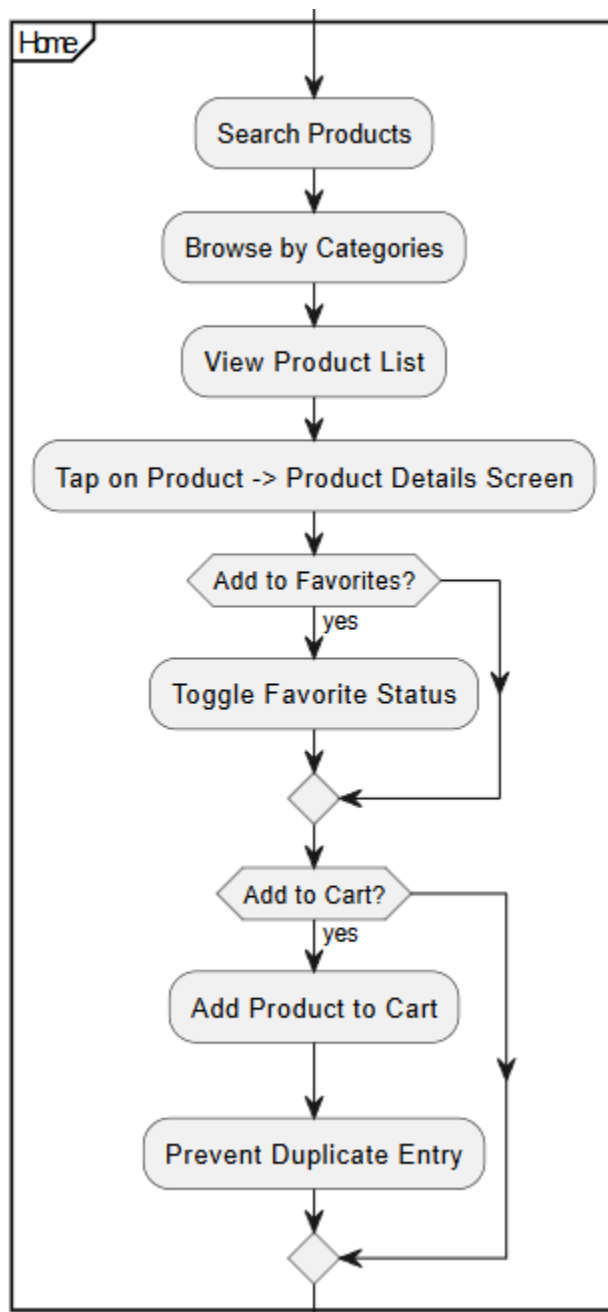


Figure: Workflow Diagram 1.3

3.2.4 Workflow Diagram: Bayer

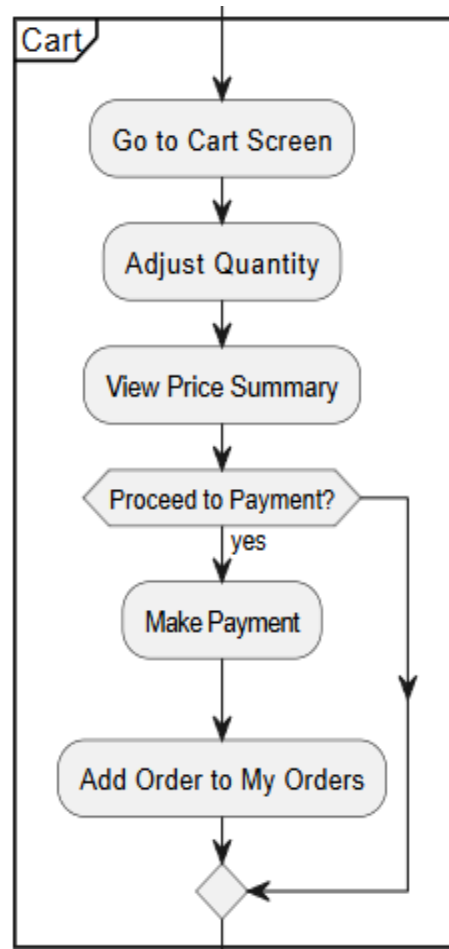


Figure: Activity Diagram 1.4

3.2.5 Workflow Diagram: Bayer

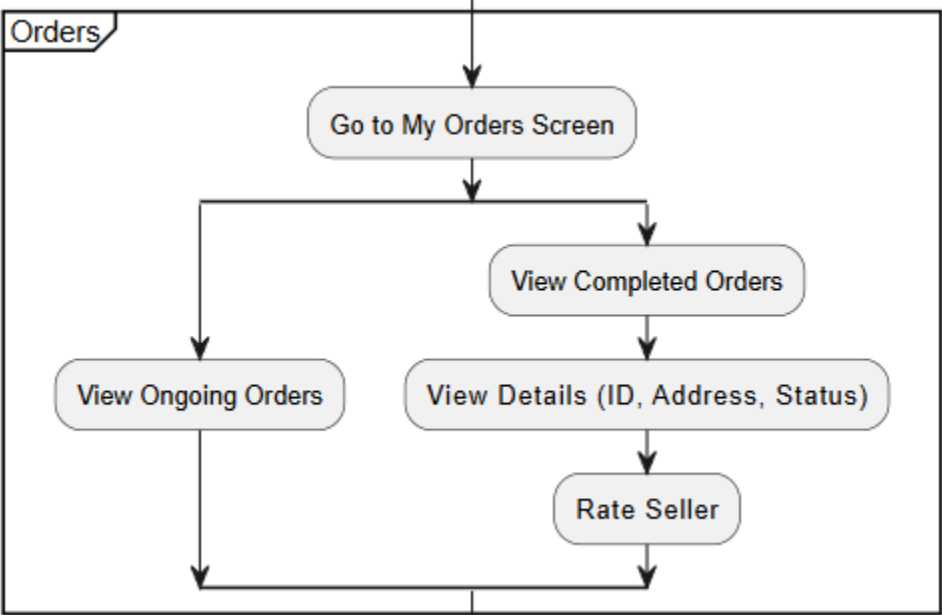


Figure: Workflow Diagram 1.5

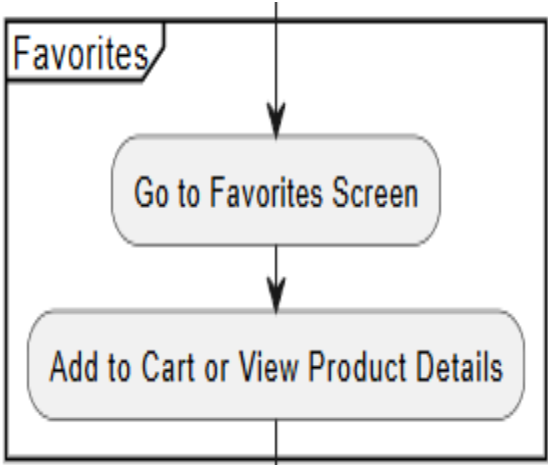


Figure: Workflow Diagram 1.6

3.2.6 Workflow Diagram: Bayer

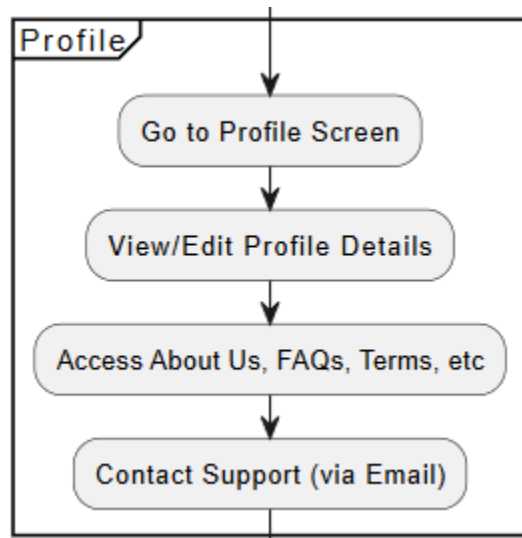


Figure: Workflow Diagram 1.7

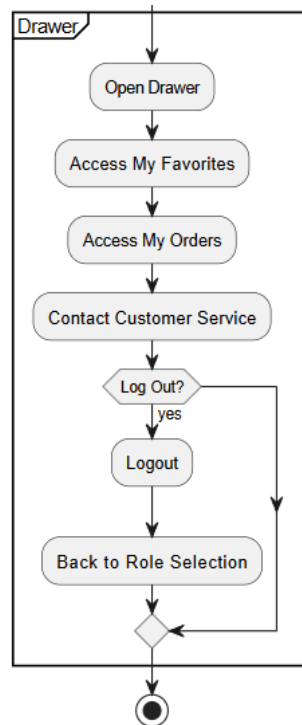


Figure: Workflow Diagram 1.8

3.2.7 Workflow Diagram: Seller

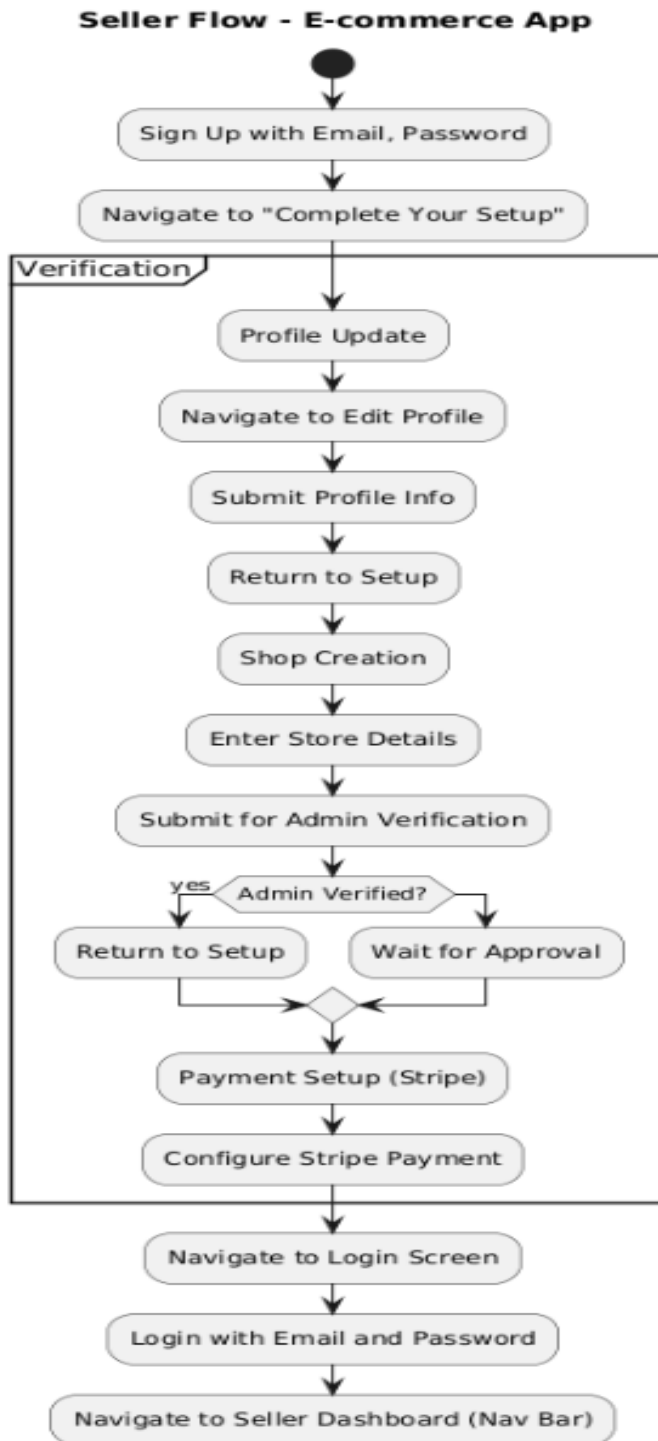


Figure: Workflow Diagram 1.1

3.2.8 Workflow Diagram: Seller

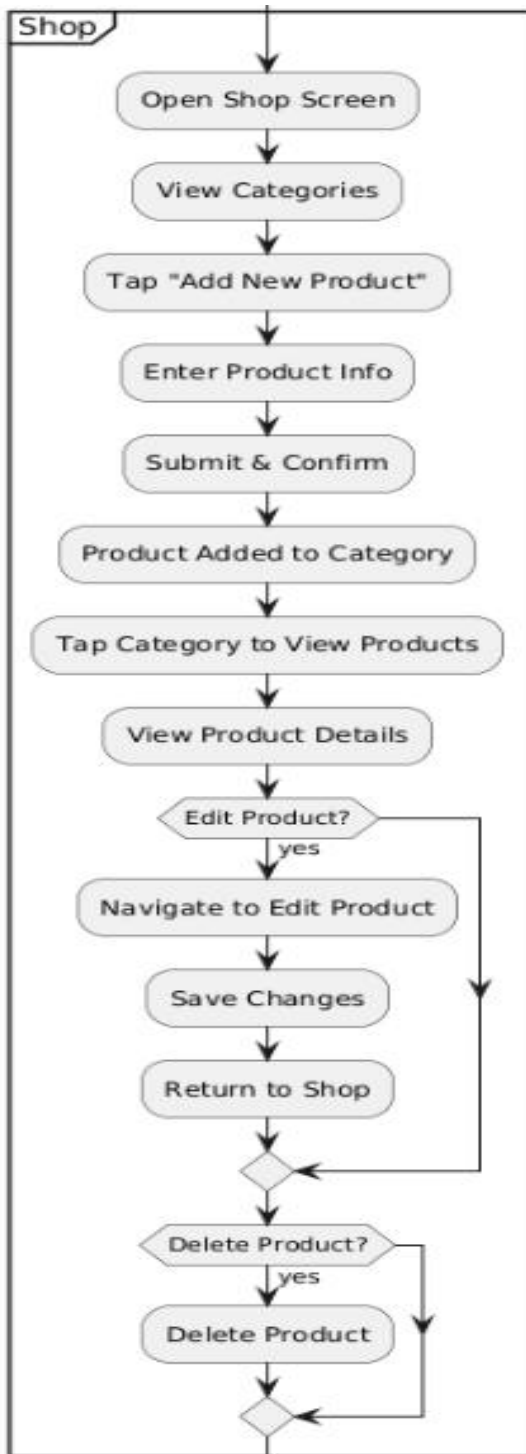


Figure: Workflow Diagram 1.2

3.2.9 Workflow Diagram: Sellers

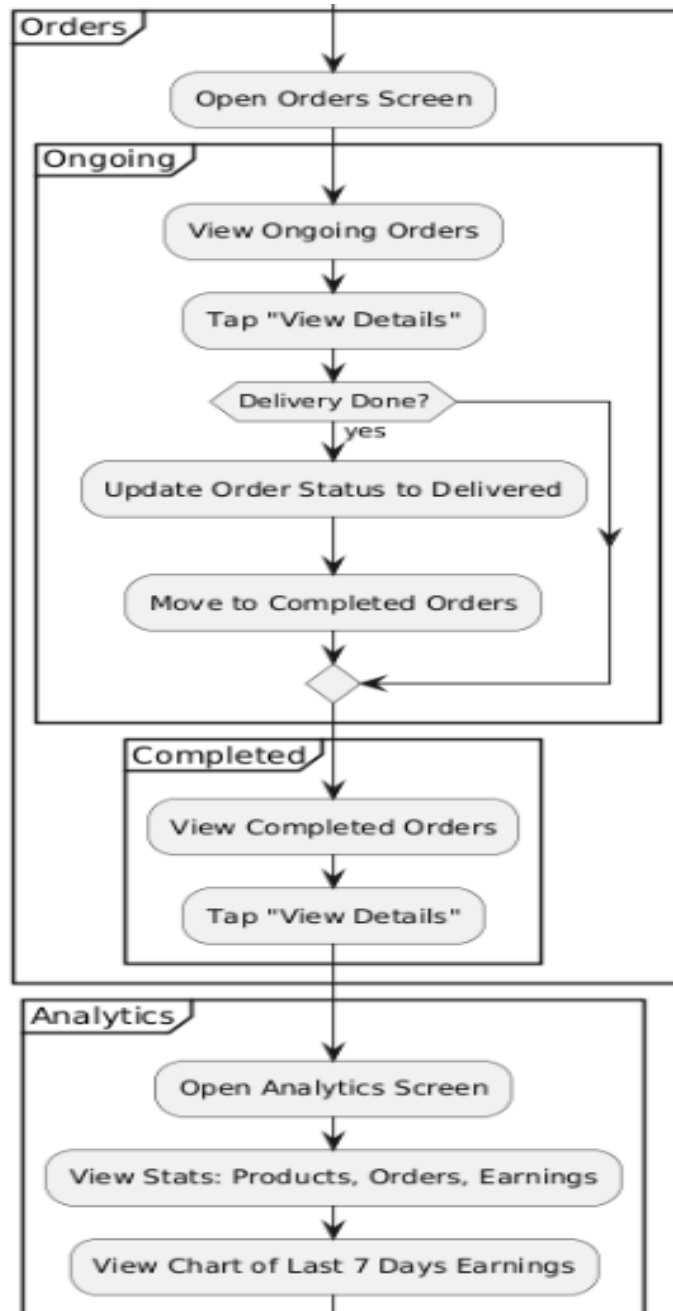


Figure: Workflow Diagram 1.3

3.2.10 Workflow Diagram: Seller

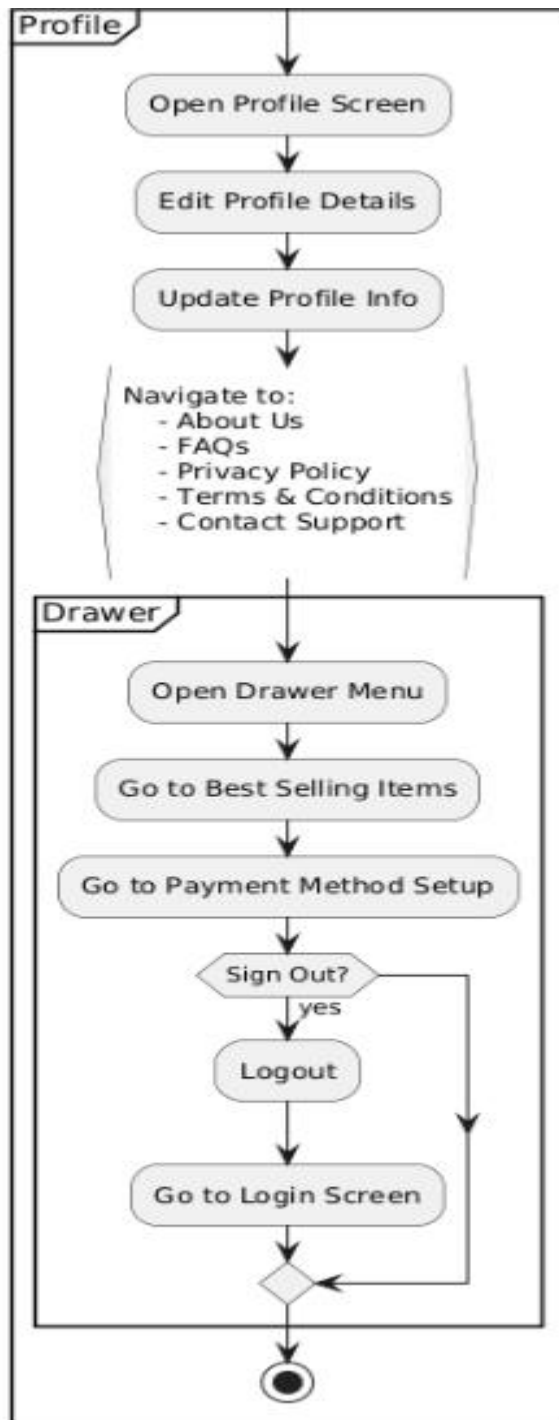


Figure: Workflow Diagram 1.4

3.2.11 Workflow Diagram: Admin

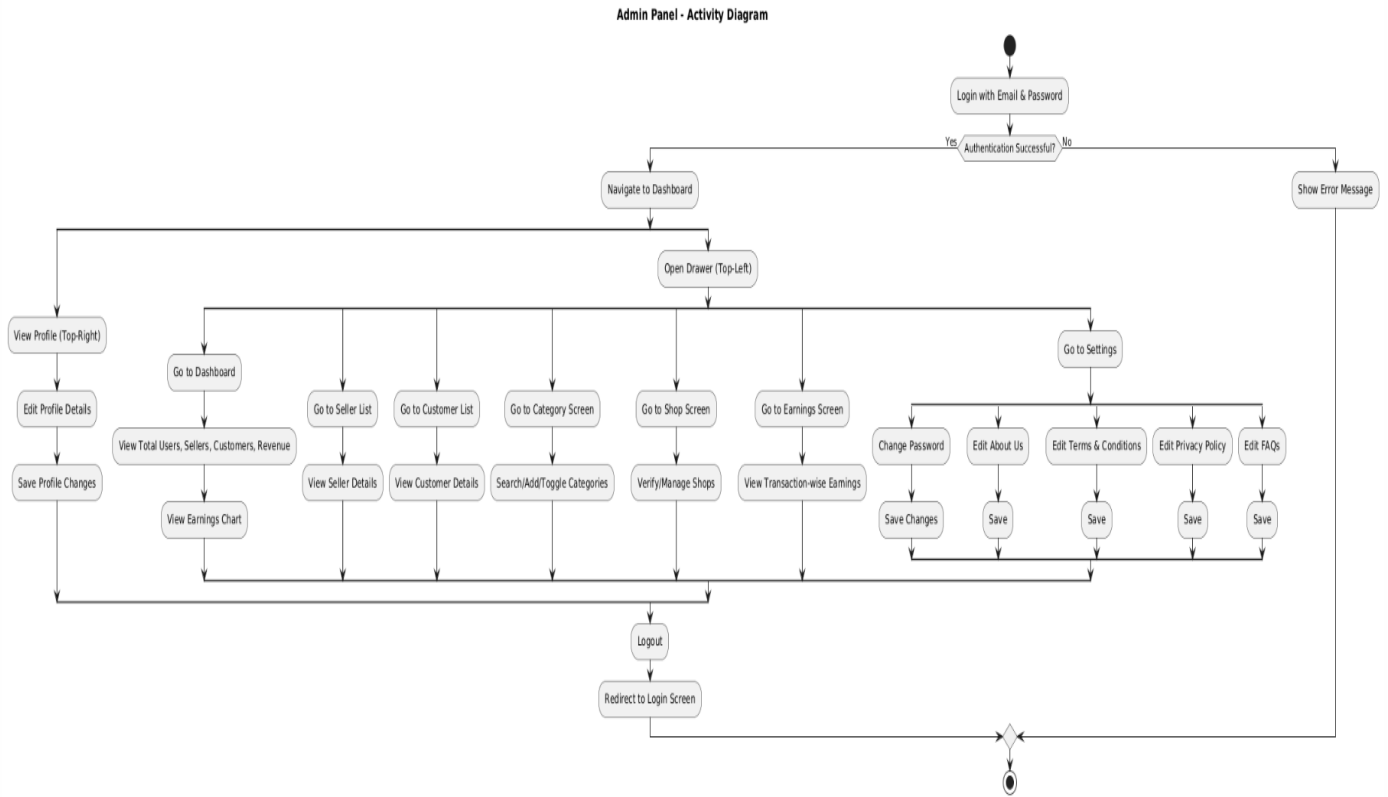


Figure: Workflow Diagram 1.1

3.3 Interaction Diagram:

3.3.1 Interaction Diagram (Buyer)

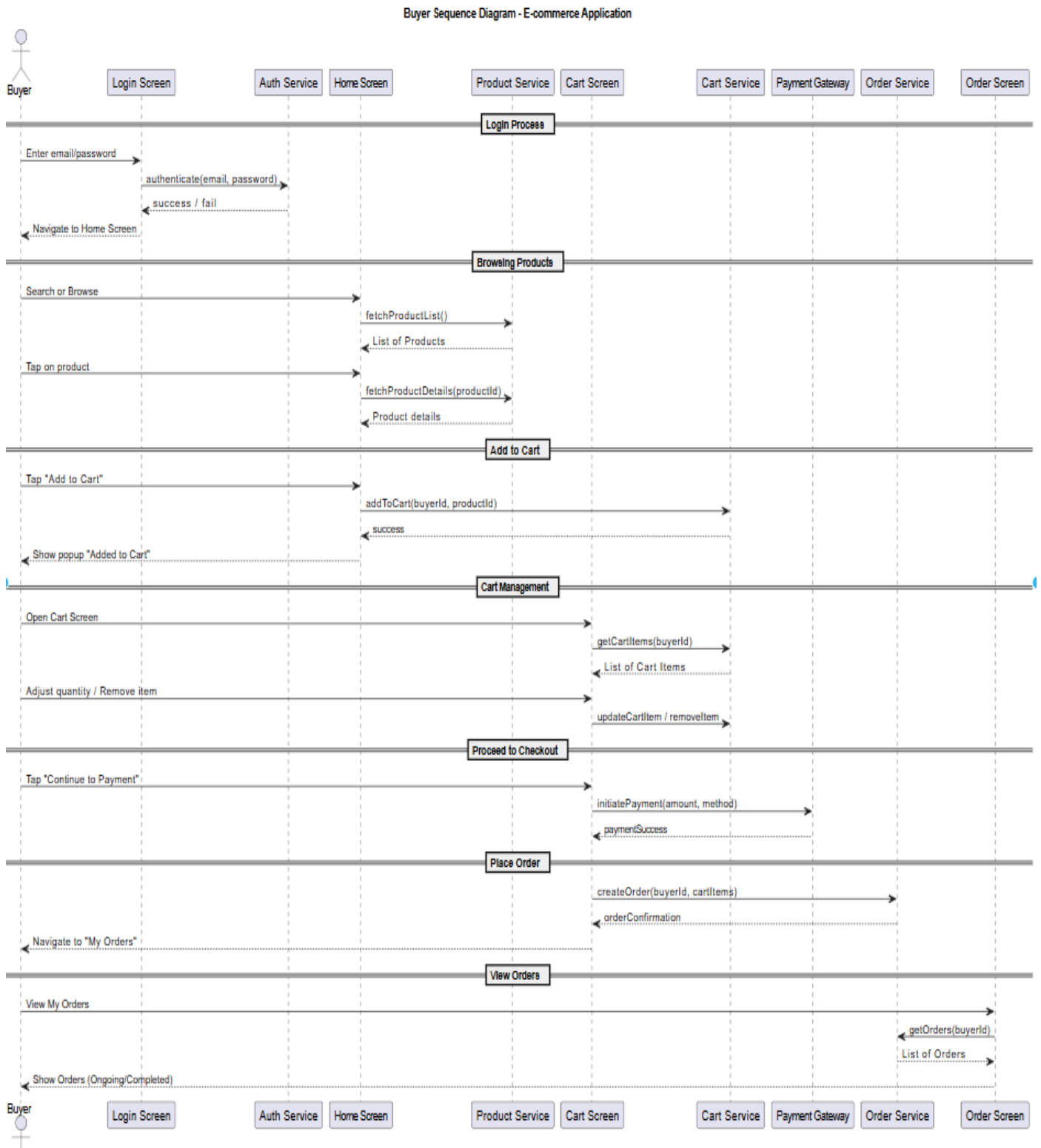


Figure: Interaction Diagram 1.1

3.3.2 Interaction Diagram (Seller)

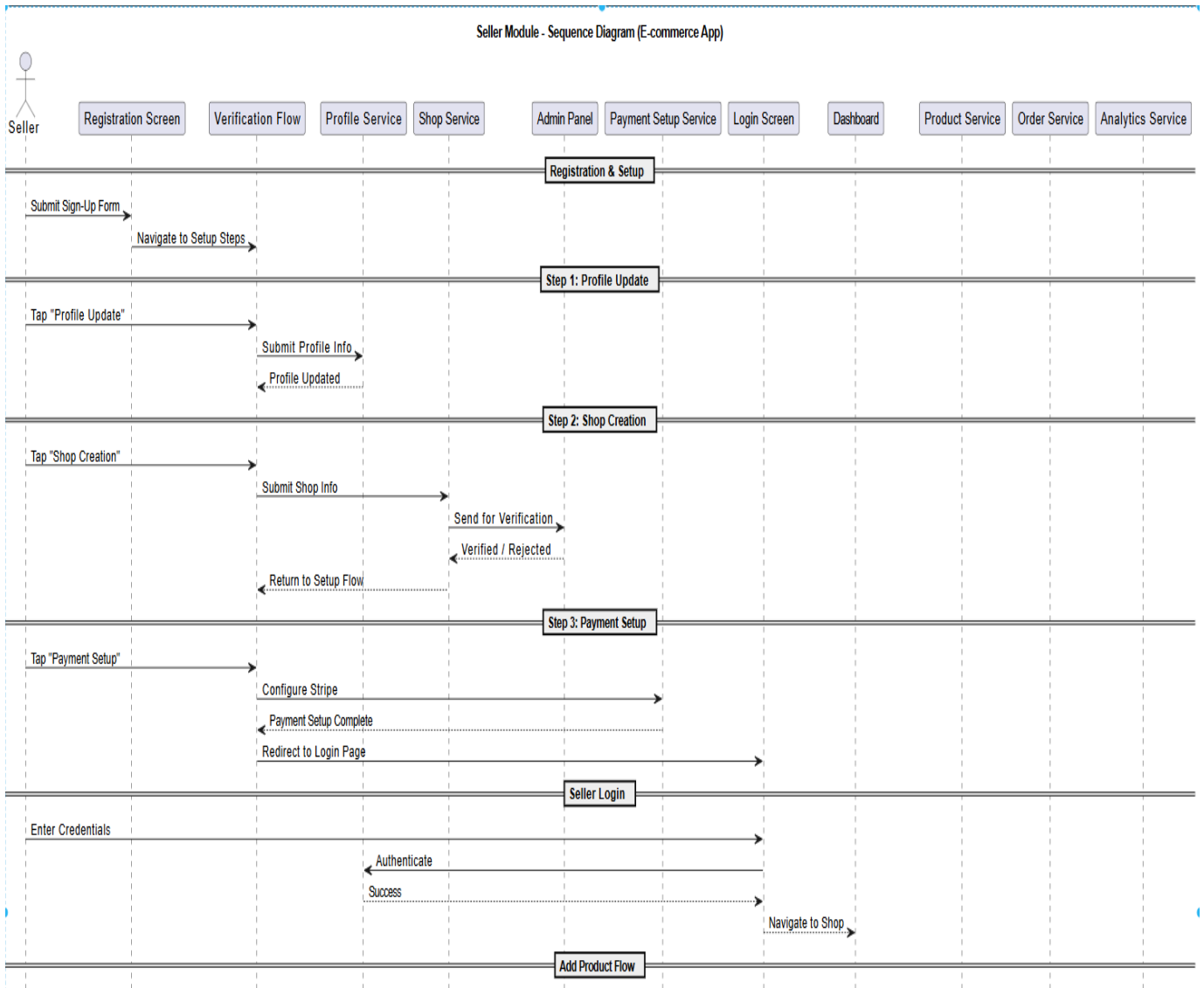


Figure: Interaction Diagram 1.1

3.3.3 Interaction Diagram (Seller)

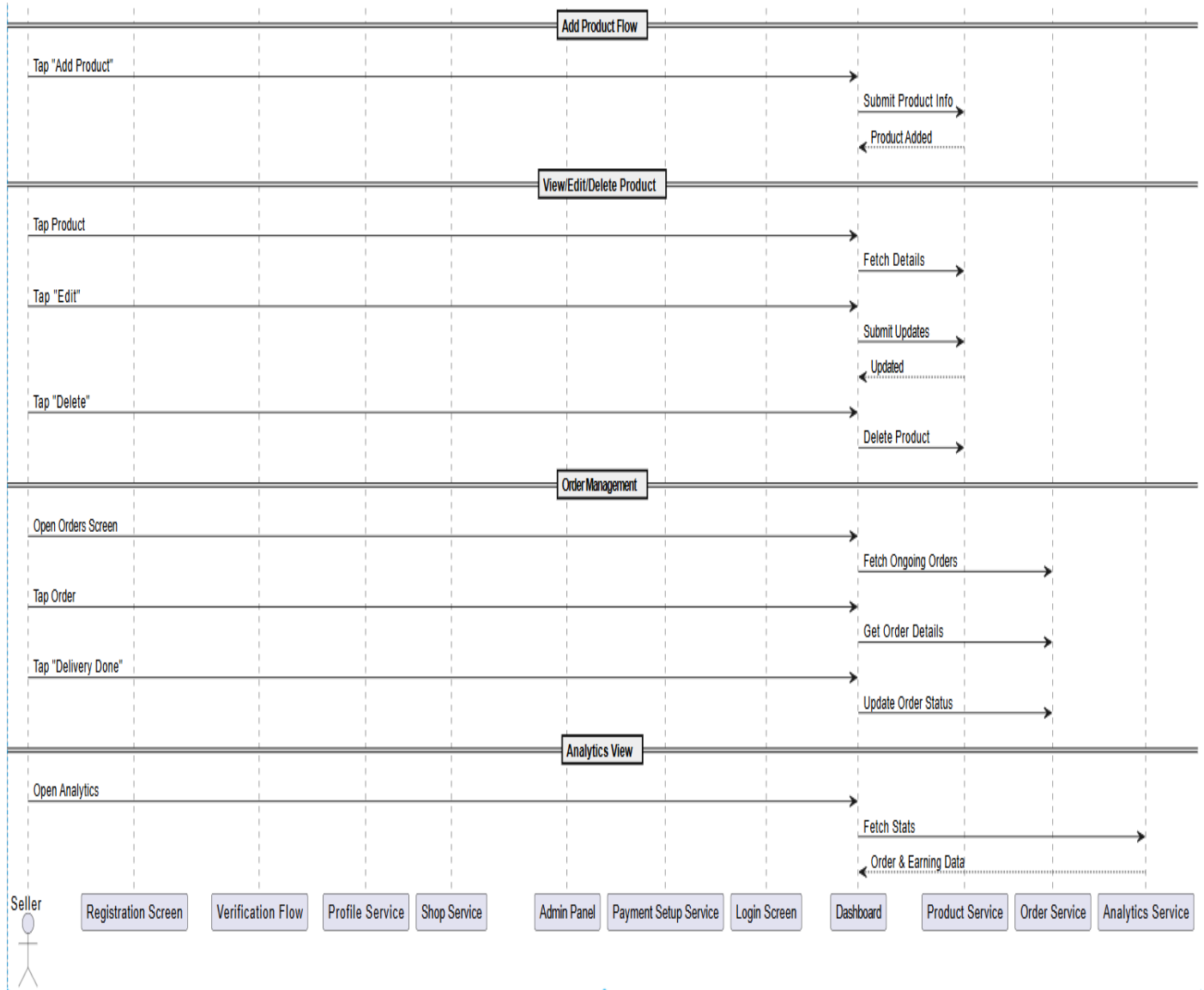


Figure: Interaction Diagram 1.2

3.3.4 Interaction Diagram (Admin)

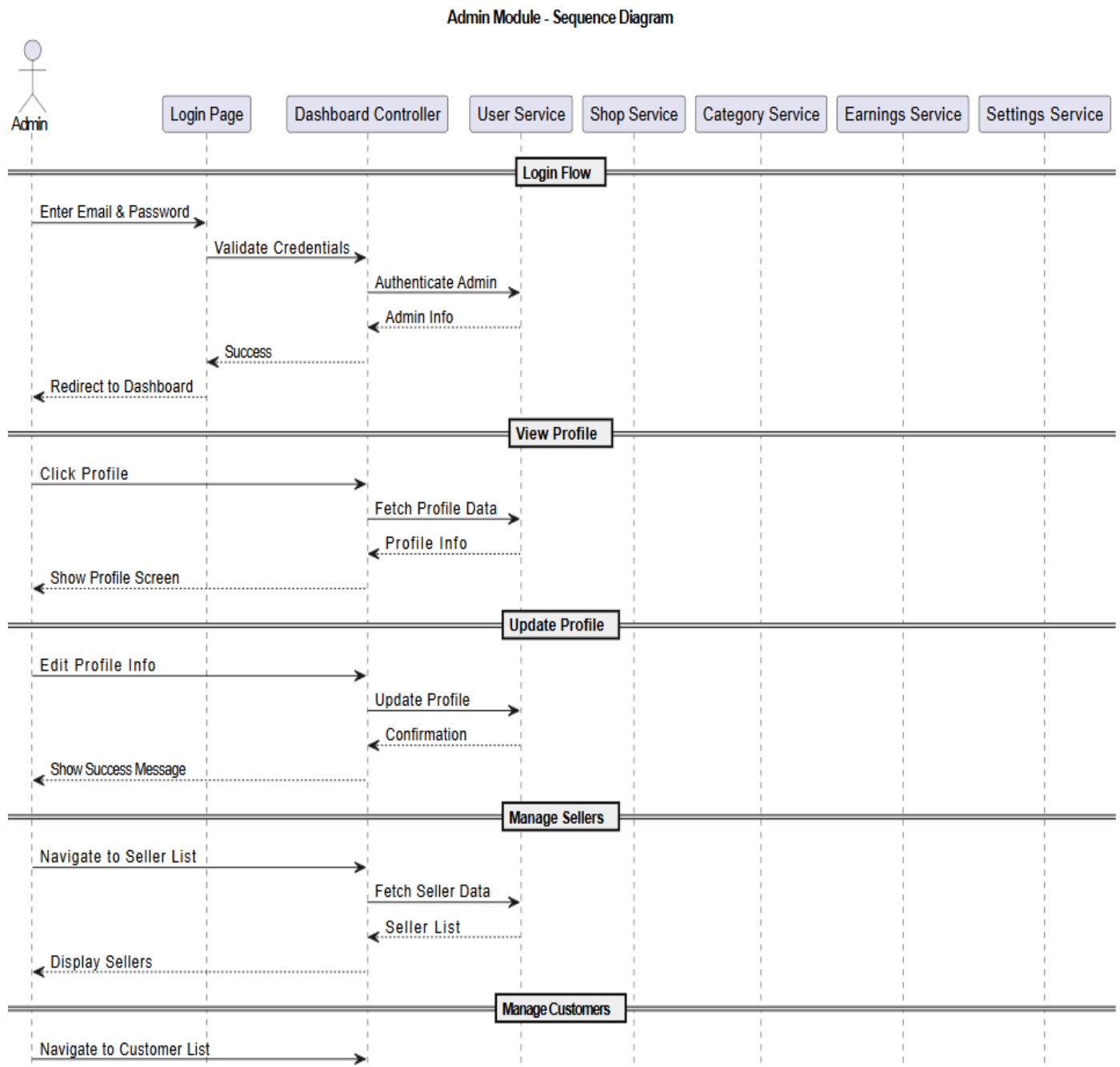


Figure : Interaction Diagram 1.1

3.3.5 Interaction Diagram (Admin)

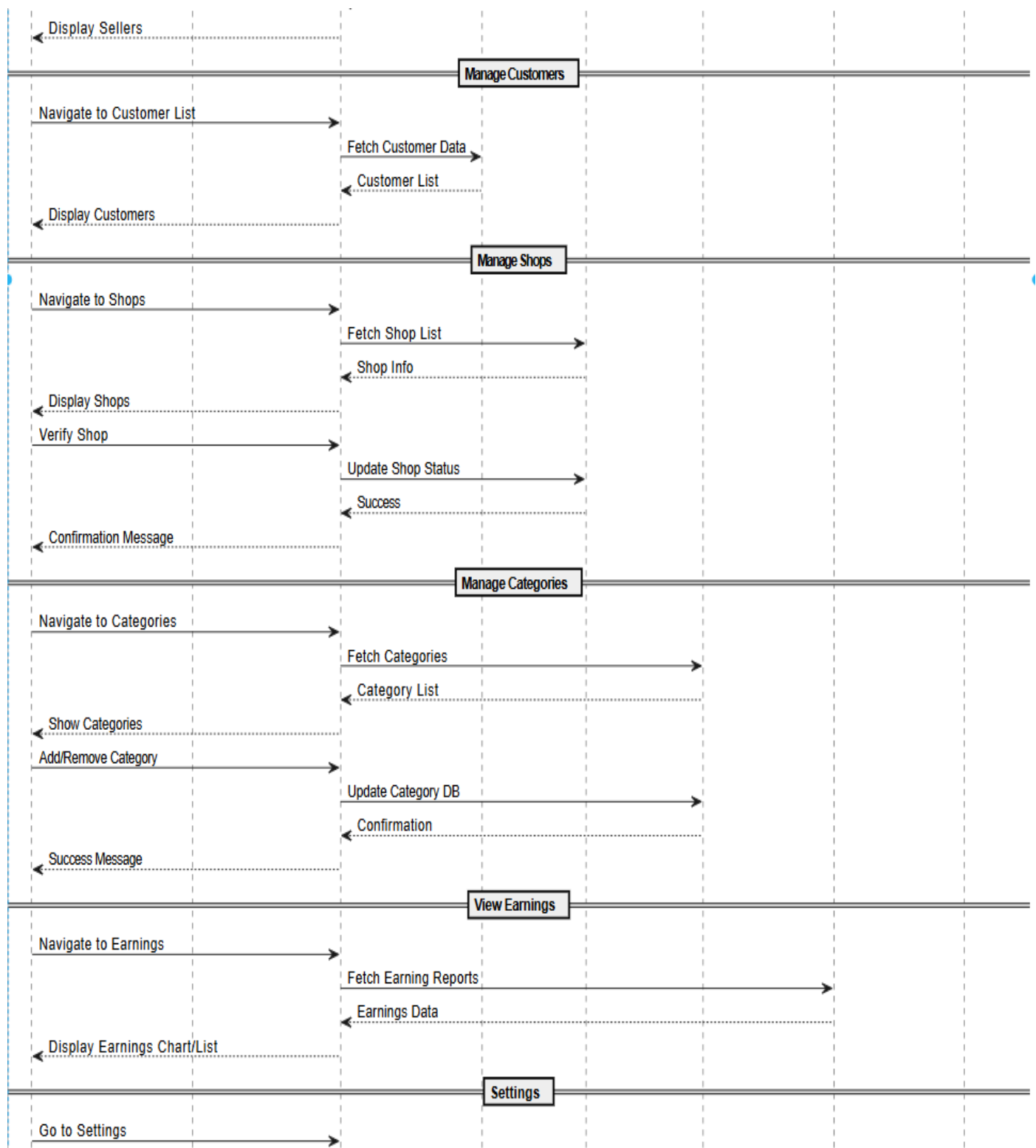


Figure: Interaction Diagram 1.2

3.3.6 Interaction Diagram (Admin)

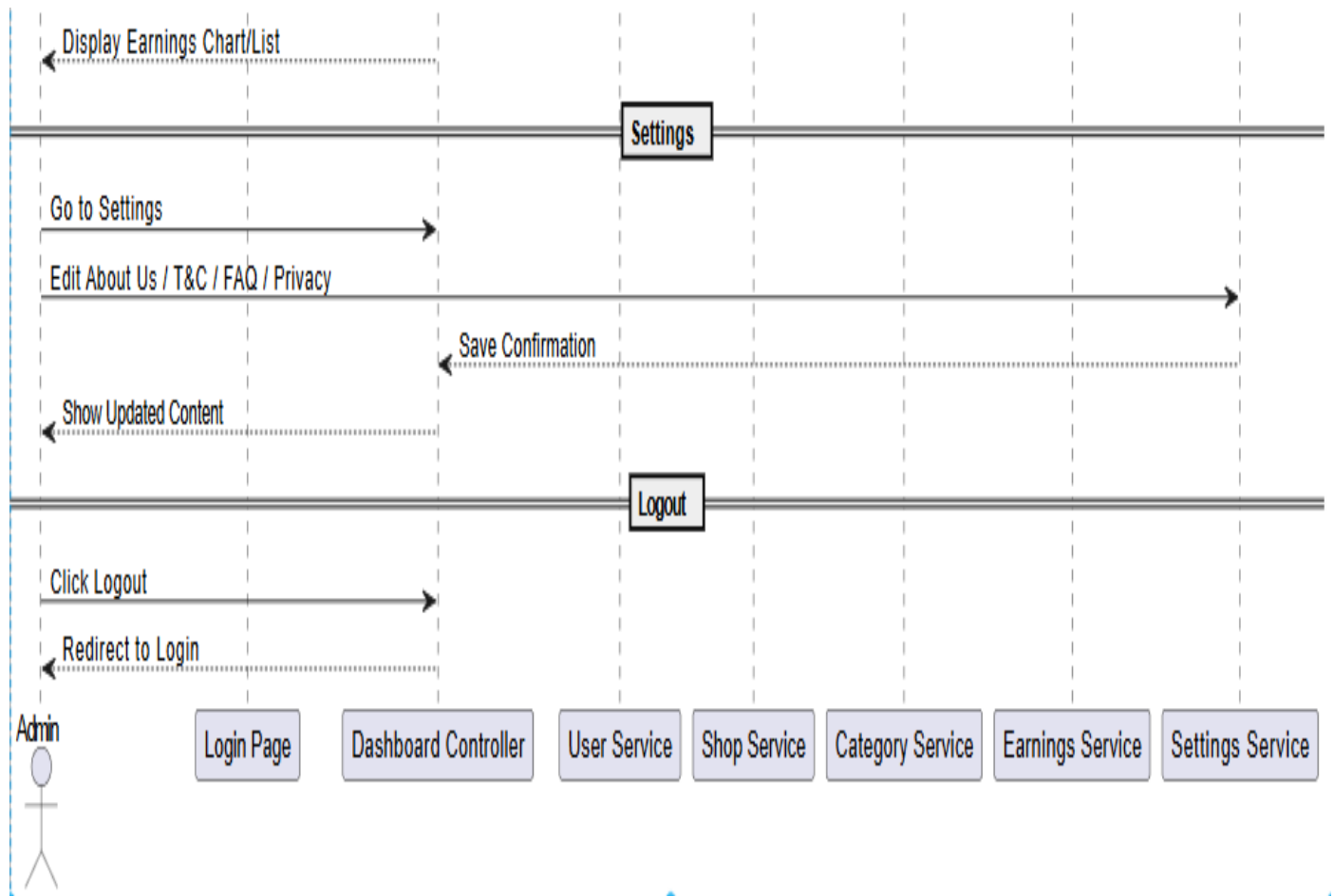


Figure: Interaction Diagram 1.3

3.4 ER Model Diagram (Buyer)

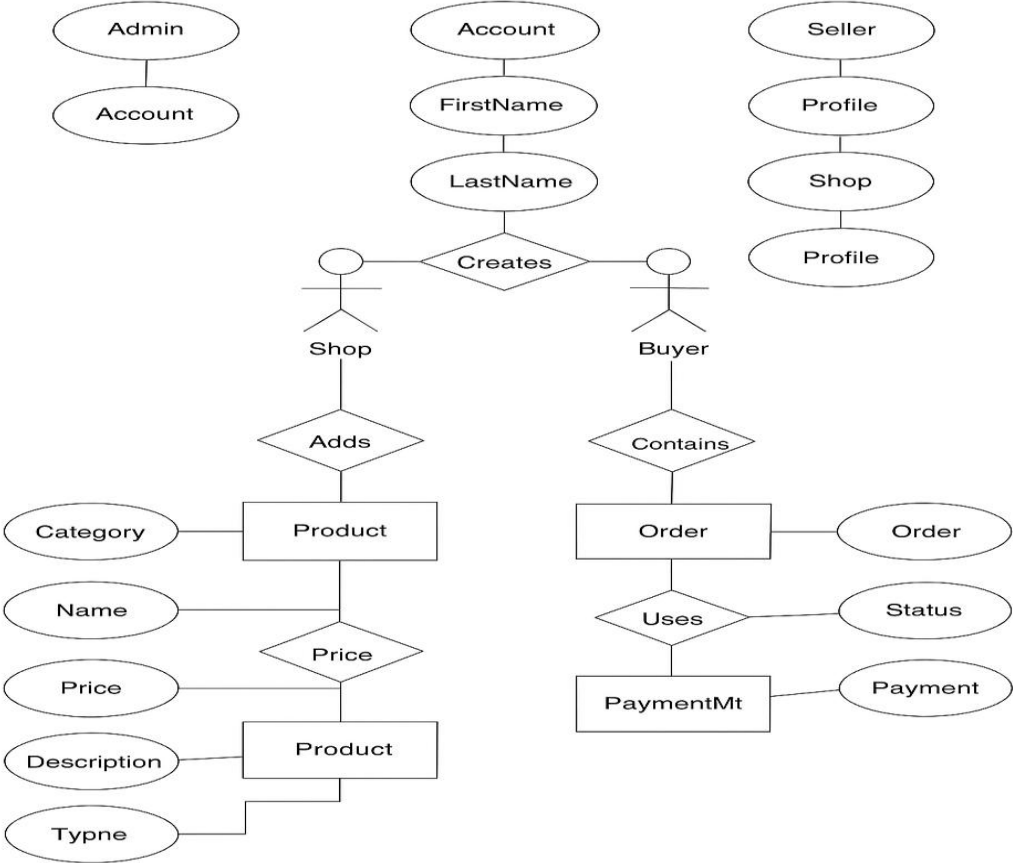


Figure: ER Model Diagram

3.5 Class Model:

3.5.1 Class Model (Buyer)

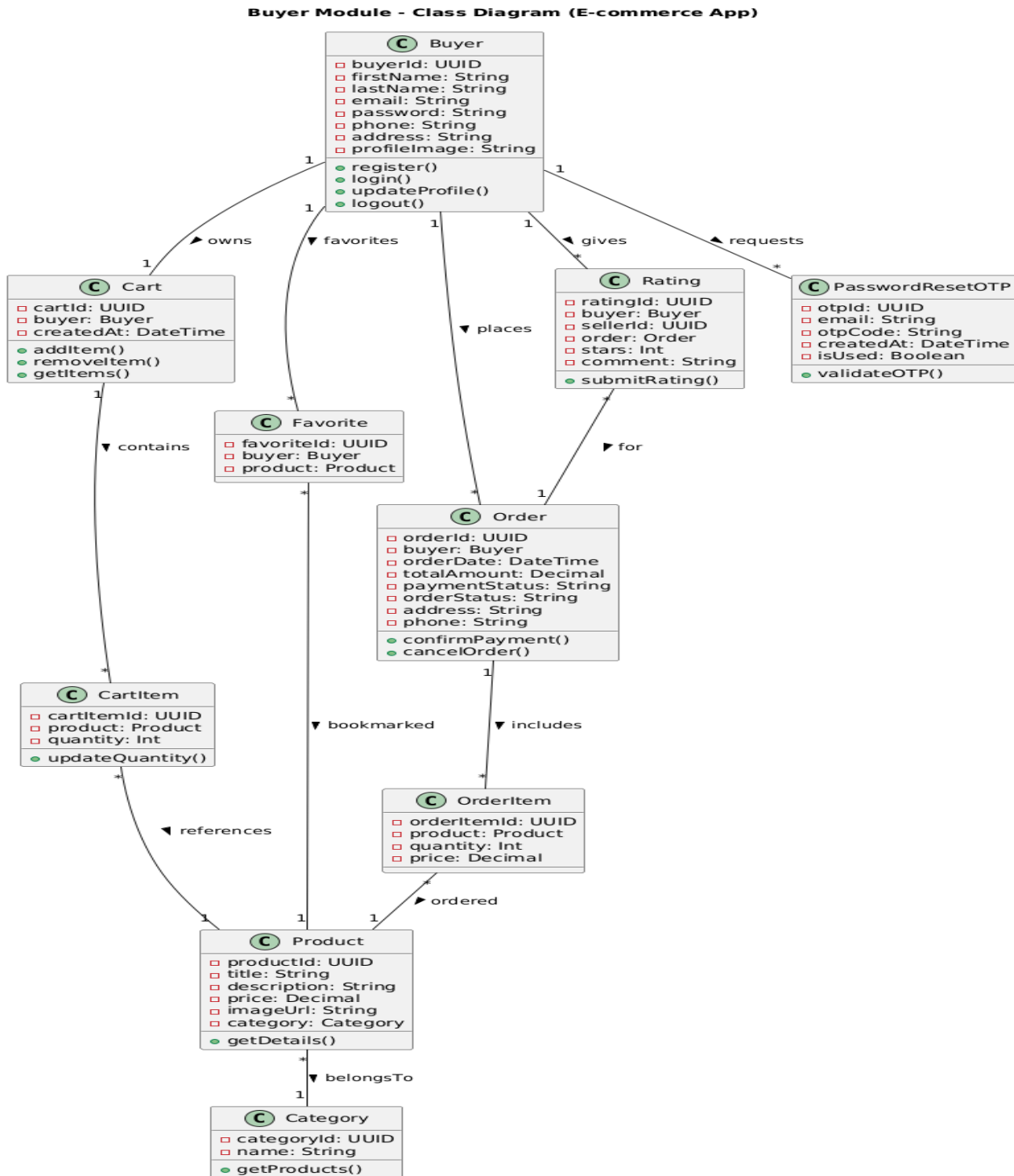


Figure: Class Model 1

3.5.2 Class Model (Seller)

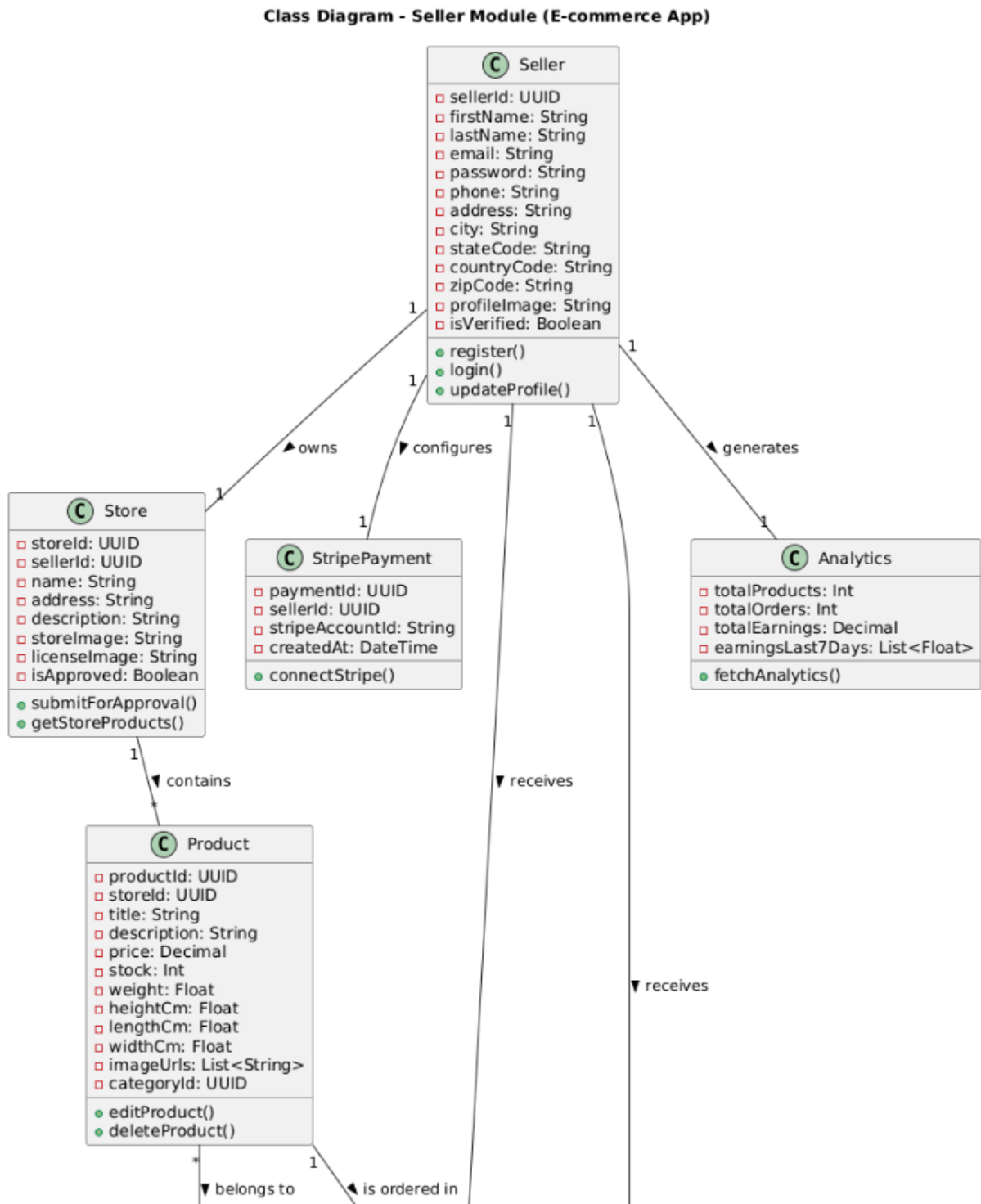


Figure : Class Model 1.1

3.5.3 Class Model (Seller)

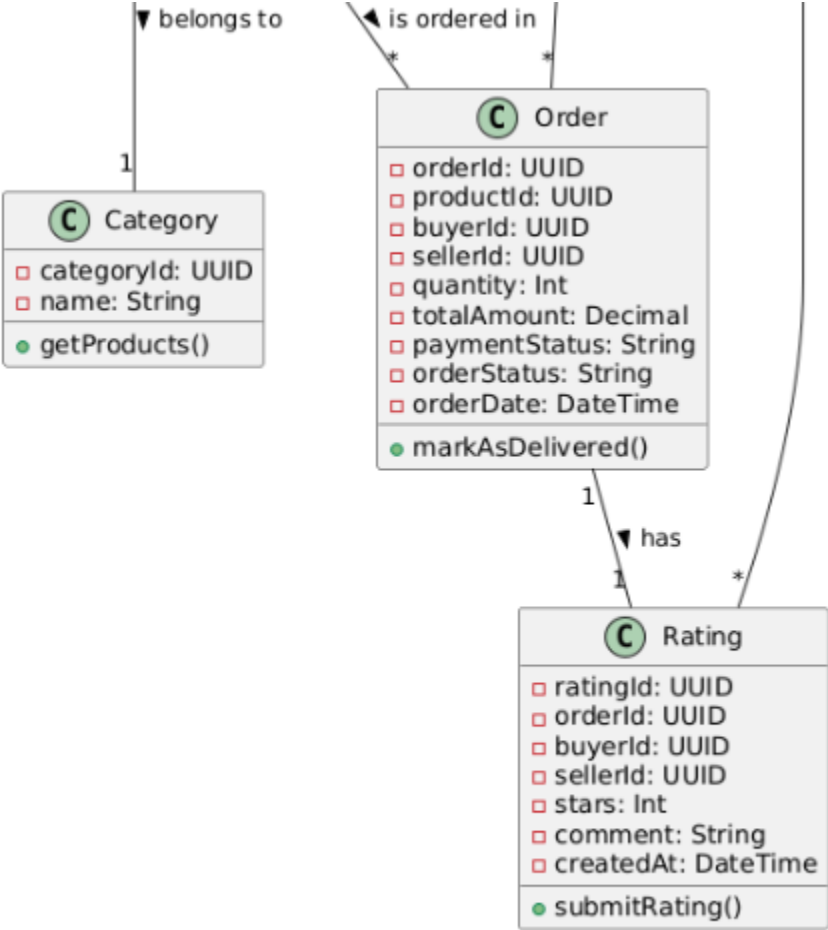


Figure : Class Model 1.2

3.5.4 Class Model (Admin)

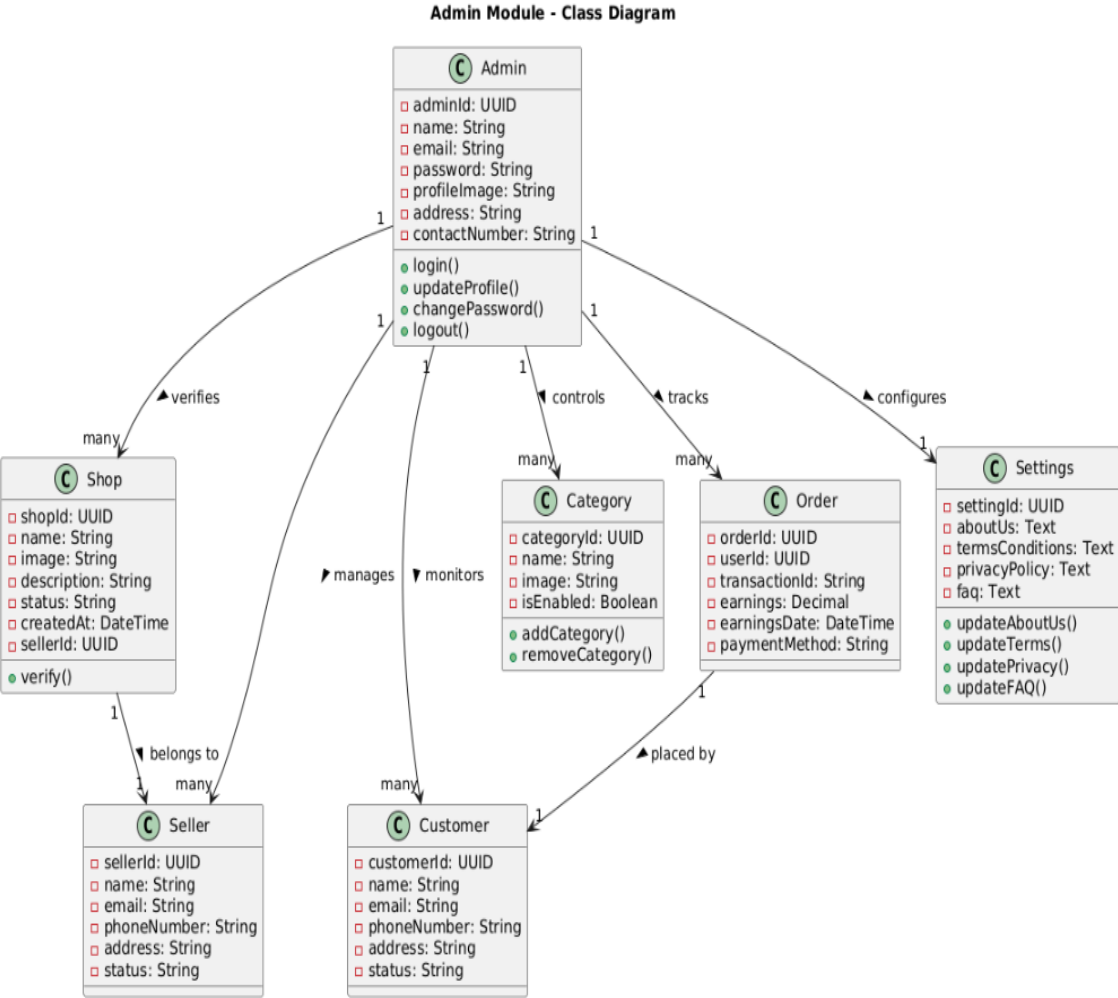


Figure: Class Model 1

3.6 System Architecture Diagram:

3.6.1 Block Diagram (Buyer)

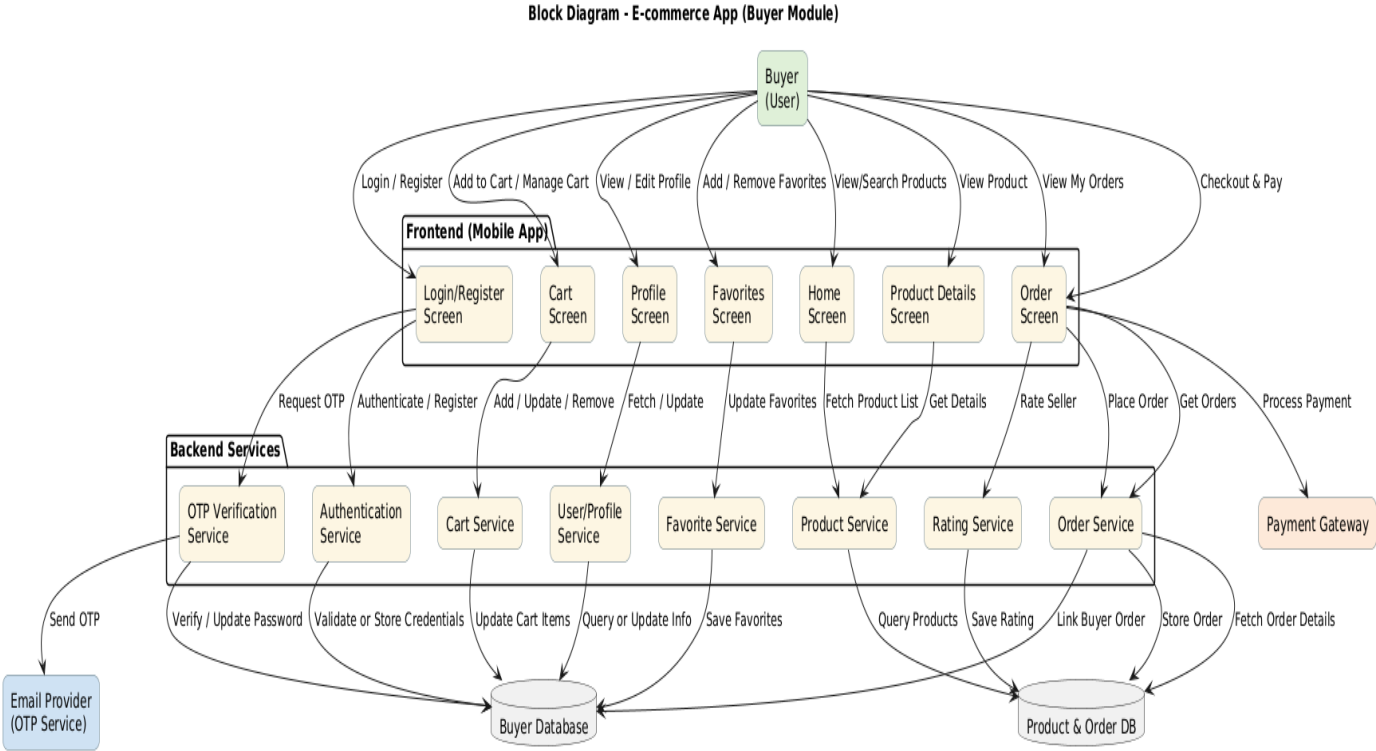


Figure: Block Diagram 1

3.6.2 Block Diagram (Seller)

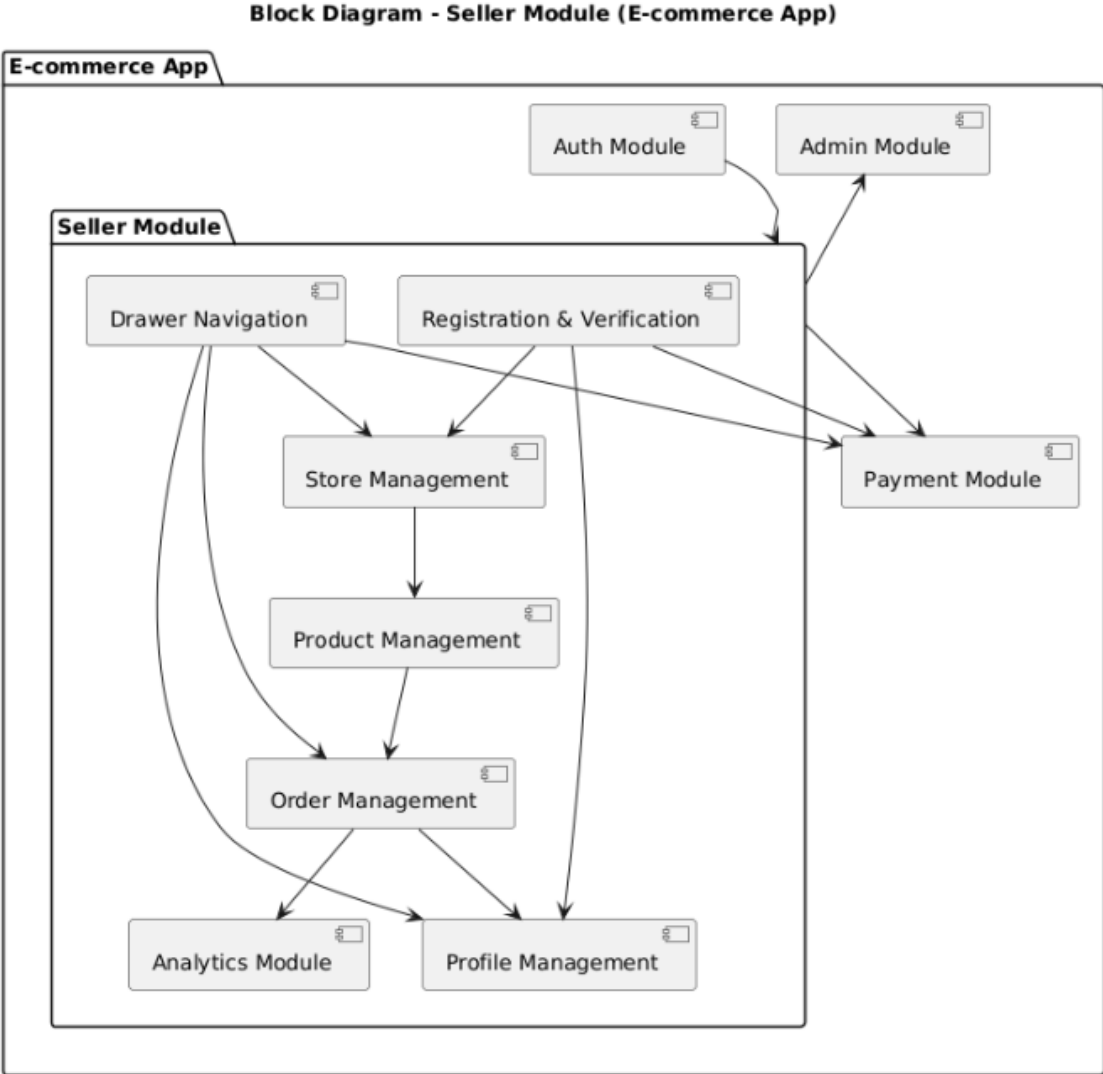


Figure: Block Diagram 1

3.6.3 Block Diagram (Admin)

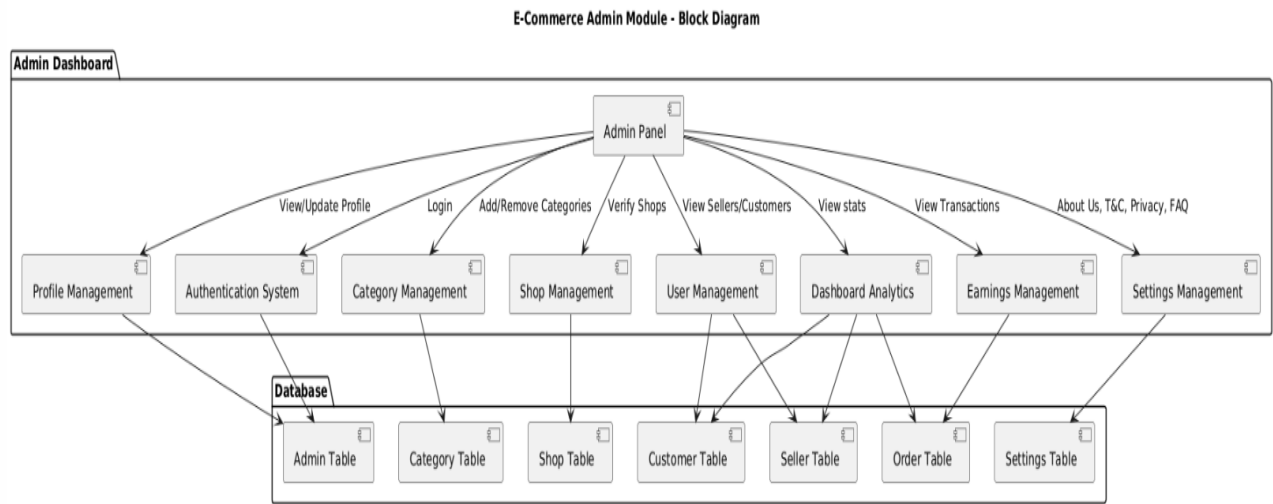


Figure: Block Diagram 1

Chapter 4: Tools and Technologies Used

4.1 Code Development Platform	VS Code
4.2 Coding Language	Dart
4.3 UI Design	Figma
4.4 Database	MongoDB
4.5 Deployment Environment	Domain

Chapter 5: Application Testing

5.1 Test Functions

System testing ensures that each feature in the application works as expected and offers a smooth unified experience for Buyers and Sellers. The testing phase targets on validating functionality, reliability, and accuracy of the core features across both user roles.

5.1.1 Feature to undergo testing

Features	Description of Priority
Log in	Buyer must log in using email and password securely.
Registration	Buyer information should be stored correctly during registration.
Browse Products	Should display a list of products by category, search, or all.
View Product Details	Should show full product details when tapped.
Add Item to Cart	The ability to add selected items to the cart should be available for users.
Place Order	Should allow the buyer to confirm and place an order.
View Orders	Must show ongoing and completed orders properly.
Add to Favorites	Products can be marked as favorite and stored.
Update Profile	Buyer must be able to update their profile info.

Figure: Feature Ranking Table for Buyer

Attributes	Description of Priority
Sign In	Seller shall log in using secure credentials.
Register	Seller profile must be stored correctly during sign-up.
Add Product	Must allow adding a new product with image and description..
Edit Product	The seller has the ability to update product details.
Discard Product	Products can be removed from their list.
View Orders	Must show all placed orders and their status.
Update Order Status	Seller can mark orders as shipped or completed.
View Earnings	Displays total earnings from sales.
Update Profile	Seller must be able to edit their personal and business info.

Figure: Feature Ranking Table for Seller.

Attributes	Priority Description
Sign In	Admin needs to log in securely using email and password.
View All Buyers	Must return a list of all registered buyers or an empty list. If none exist.
View All Sellers	Must return a list of all registered sellers or an empty list if non exist.
Approve Seller	Admin should be able to verify and approve seller accounts.
Remove Buyer	Admin can delete or disable a buyer account if necessary.
Remove Seller	Admin can delete or disable a seller account if necessary.
View All Products	Should display all products across the platform.
Remove Product	Admin must be able to remove any inappropriate or reported product.
Manage Reports	Must allow handling of complaints or reports from users (e.g., against sellers/products).
View Platform Stats	Should show analytics (total users, total sales, popular products, etc.).
Manage FAQs & Policies	Admin must be able to update Privacy Policy, Terms & Conditions, About Us, etc.
Contact Support Replies	Admin should view and respond to support/contact

Figure: Features priority table for Admin.

5.1.2 Evaluation Strategies

5.1.3 Test Technique:

- Practical testing, or "black-box" testing
- White-box inspections, also known as structural testing

To maintain quality, reliability, and usability of the e-commerce application, I applied two major types of tests. Tests are based on **requirements and expected functionality**.

For this e-commerce app, the following were tested using black-box techniques:

- **Functional Testing**

BuyerModule:

- Account creation, login, and OTP-based password reset.
- Adding products to cart, placing orders, viewing order status, adding favorites.
- Viewing and updating profile, navigating through settings and categories.

Seller Module:

- Registration, login, product uploads, and order tracking.
- Viewing analytics for ongoing/completed orders.

Admin Module:

- User management (approve/reject sellers, view buyers), manage platform content like FAQs, Privacy Policy, Terms & Conditions.
- Monitor and remove inappropriate products.

Structural Testing

The software architecture's internal logic and operations flow were verified through the use of structural testing. This includes control flow, data flow, loops, and condition coverage.

Key areas tested:

Authentication Logic:

Verification of password hashing, token generation, and validation flow.

Order Processing:

Ensuring that product stock updates correctly and orders transition between statuses.

Seller Approval Mechanism:

Confirming that only verified sellers can upload products.

Access Control:

Ensuring that different user roles (Buyer, Seller, Admin) have proper access restrictions.

5.1.4 Test Plan Timeline

Test Stage and Timeline
Test Strategy Creation - 7 days.
Module Testing cycle
Component Evaluation - During development cycle
Module Integration Testing – 7 Days
Interface Testing – 1-Week Schedule
Performance Verification – 7 Days
Penetration Testing – 7 Days
Client/User Validation Testing – 7 Days
Accessibility Testing - 7 days.
Bug Fix & Final QA Review - 1 Week

5.2 Test Cases:

5.2.1 User Login and Access Management Control:

Validation Scenario: Buyer Login

Procedures:

- Start the app.
- Fill in a valid email and password.
- Use the login button to proceed.

Expected Result:

- Buyer is successfully logged in and navigated to the Home screen.

Test Case: Invalid Login Attempt

Steps:

- Open the app.
- Type an incorrect email or password.
- Select the login button.

Assumed Result:

- Login fails with an appropriate error message.

5.3.2 Functionality

Test Case: Add Product (Seller)

Steps:

- Log in as a Seller.
- Navigate to "Add Product" screen.
- Fill out product details and upload an image.
- Select "Submit".

Desired Result:

Product operation was successful listed and shown in the product list.

Test Use Case: Add to Cart (Buyer)

Steps:

- Sign in as a Buyer
- Browse and pick a product
- Tap "Add to Cart".

Predicted Outcome:

The shopping cart updates with the new total value when a product is added.

Performance

Use Case for Testing: High Traffic Load

Steps:

- Simulate multiple users logging in and browsing products at the same time.

Expected Result:

App remains responsive and functions without crashing.

Test Case: Checkout Response Time

Steps:

- Complete checkout process as Buyer.
- Measure response time from placing order to confirmation.

Expected Result:

Order should be placed within 2–3 seconds.

5.3.3 Usability

Test Case: Navigation Simplicity

Steps:

- Explore various sections using the bottom navigation or drawer.

Expected Result:

Navigation should feel smooth, intuitive, and consistent.

Test Case: Responsive Design

Steps:

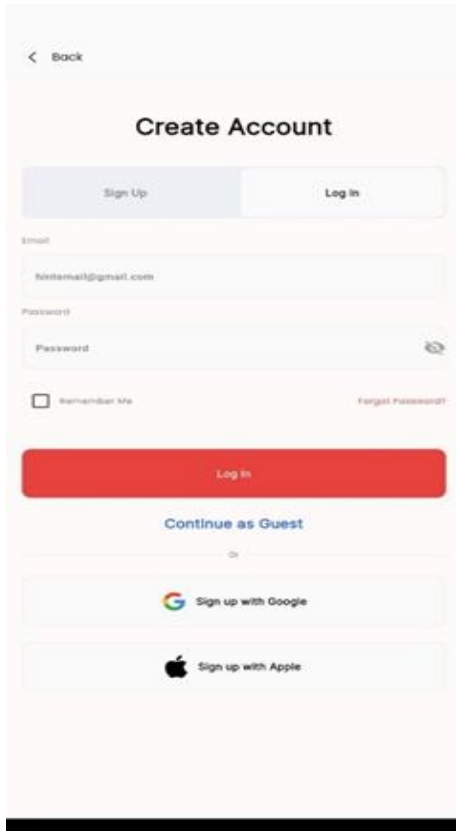
- Use the app on various screen sizes.

Expected Result:

UI elements should adjust properly and maintain usability across devices.

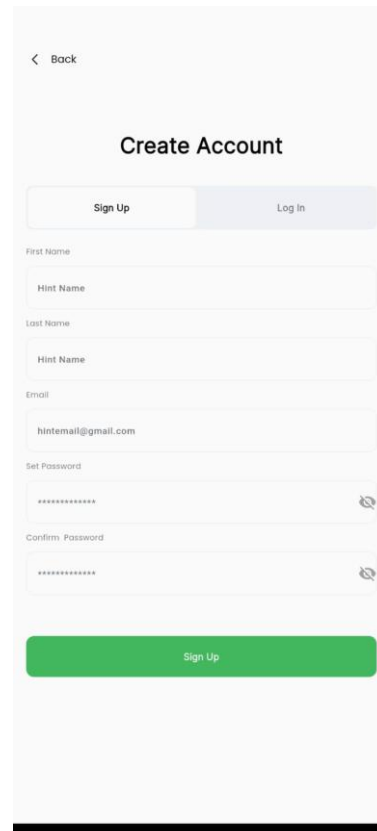
Segment 6: Operation Guide

6.1 User Login Screen:



The login screen features a 'Create Account' title and two buttons: 'Sign Up' and 'Log In'. Below these are input fields for 'Email' (containing 'hintemail@gmail.com') and 'Password'. A 'Remember Me' checkbox and a 'Forgot Password?' link are present. A prominent red 'Log In' button is centered below the inputs. Below this is a 'Continue as Guest' link, followed by social login options for Google and Apple.

6.2 User Registration Screen:

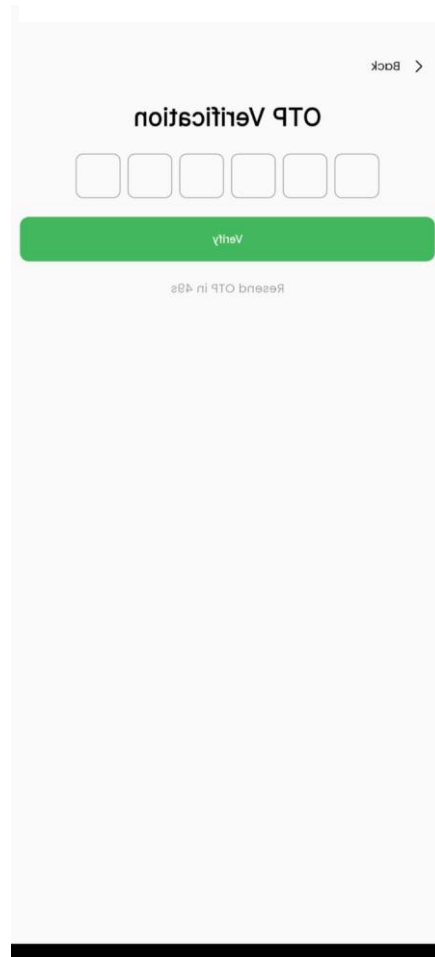


The registration screen features a 'Create Account' title and two buttons: 'Sign Up' and 'Log In'. Below these are input fields for 'First Name', 'Last Name', and 'Email' (containing 'hintemail@gmail.com'). There are also 'Hint Name' fields for both first and last names. Below the email field are 'Set Password' and 'Confirm Password' fields, both masked with asterisks. A green 'Sign Up' button is centered at the bottom of the form.

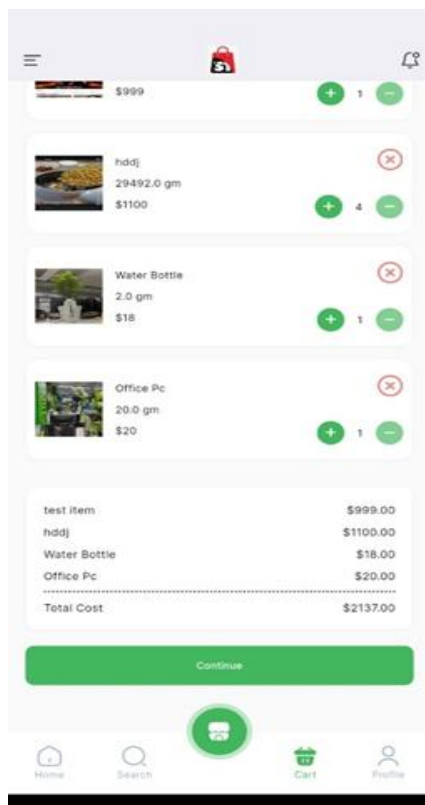
6.2 Role Selection Page:



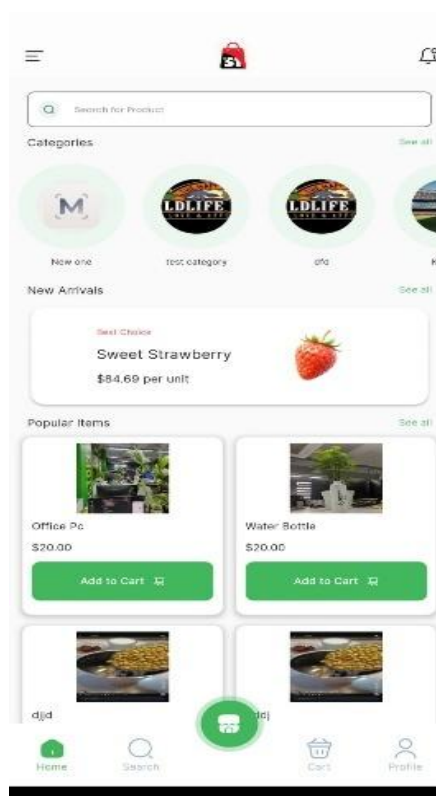
6.1 OTP Verify Page:



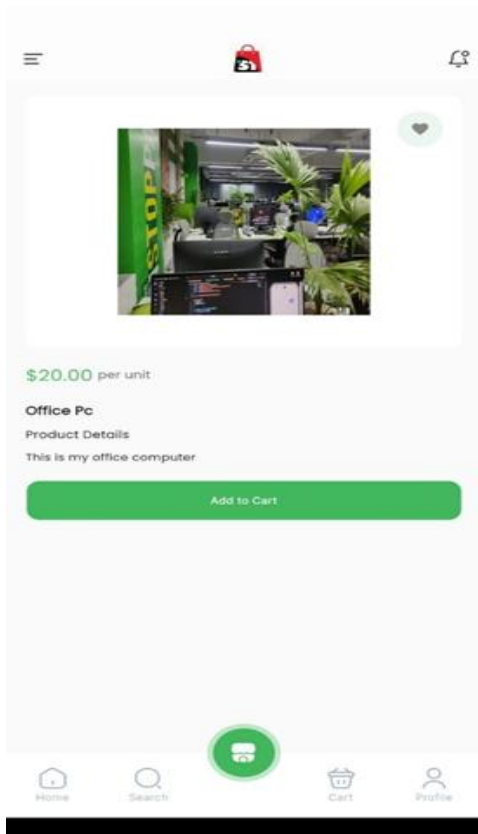
6.2 Cart Page:



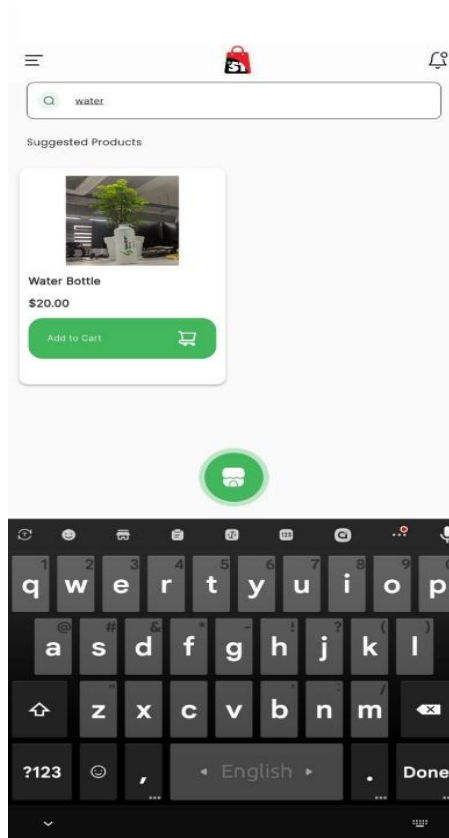
6.1 Home Page:



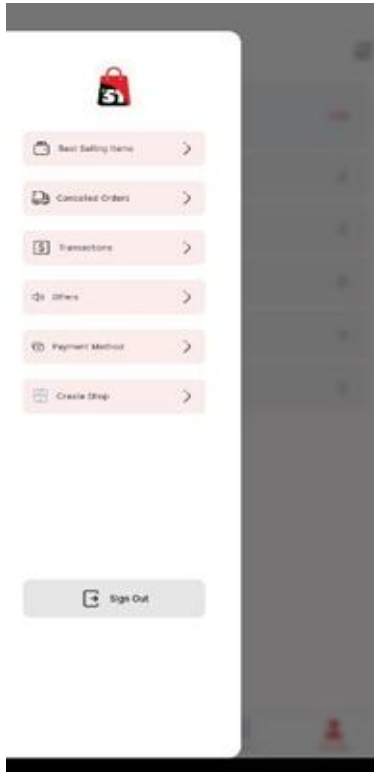
6.4 Product Details Page:



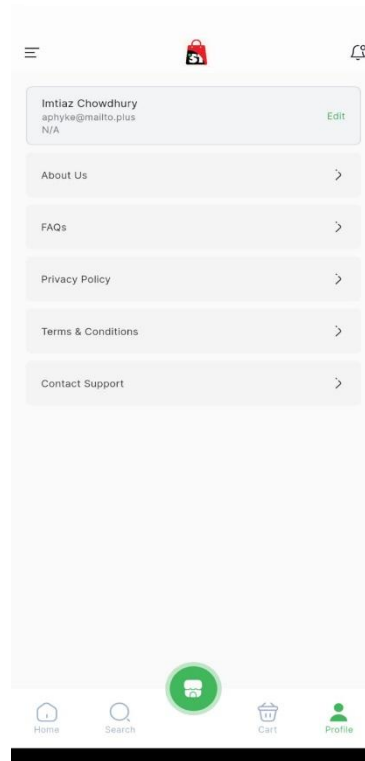
6.3 Search Page:



6.2 Seller Drawer Screen:



6.1 Profile Page:



6.2 Personal Information Page :

Edit Profile

Account Details

User Name
Imti az

Phone Number
01626583370

Address Line 1
Dhaka Rampura

Address Line 2
Hint Address

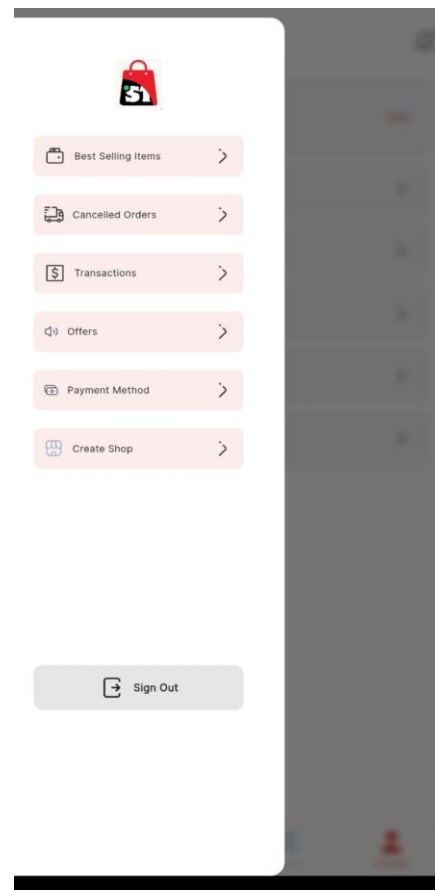
Enter Your City
Dhaka

Enter Country Code
880

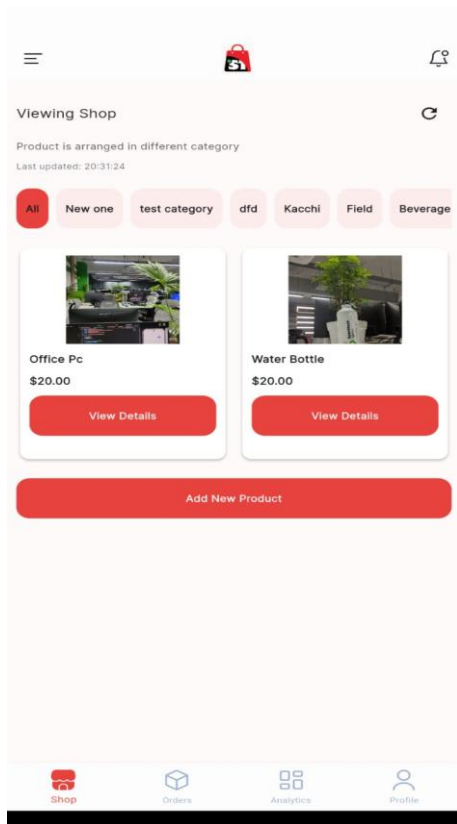
Enter Your State Code
1204

Enter Your Zip Code
1204

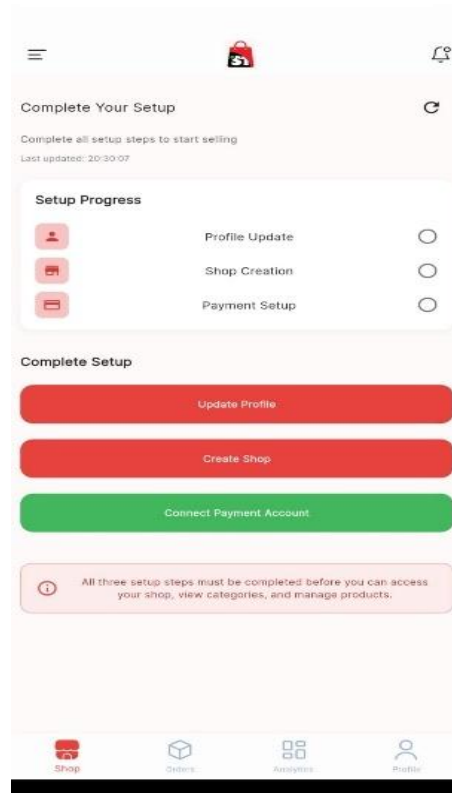
6.1 User Manager:



6.2 Seller Account Page:



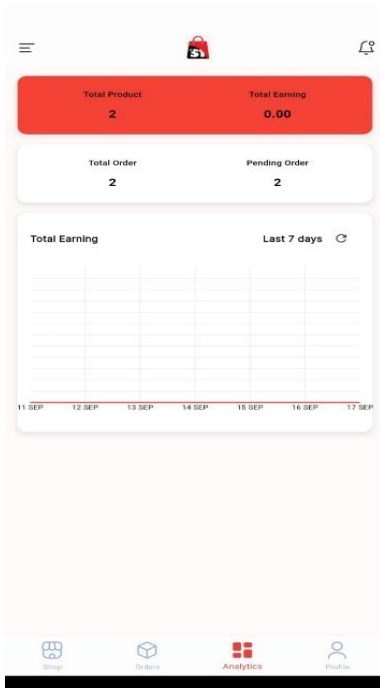
6.1 Seller Verify Shop Screen:



6.1 Vendor Shop Screen:



6.2 Seller Analytics Screen:



6.1 Seller Profile Screen:

The Seller Profile screen displays the following information:

- Inti az**
intiaz@hallo.plus
01929563370 [Edit](#)
- [About Us](#)
- [FAQs](#)
- [Privacy Policy](#)
- [Terms & Conditions](#)
- [Contact Support](#)

Navigation icons at the bottom: Home, Orders, Analytics, Profile (highlighted).

7.1 Project Access URL

[Click Here](#)

7.1 Drawbacks:

- **Real-time Chat** between buyers and sellers is not available.
- **Order update and promotional push notifications** are not included in this version.
- **Advanced search filters**, such as price range and rating, have limited functionality.
- **Admin Analytics Dashboard** provides only basic reporting features

7.3 Future Scopes:

In future updates, several improvements can enhance the system:

- **Real-time Chat Support** between buyers and sellers can improve communication.
- **Advanced Analytics Dashboard** for sellers and admins can help monitor sales trends and customer behavior.
- **AI-Based Product Recommendation** can be implemented to improve user experience.
- **Multi-language Support** may be introduced to serve a wider audience.

Cloud Backup for data security and easy recovery will be considered in future versions.

Reference:

1. **Ian Sommerville, Software Engineering, 10th Ed., 2015**
2. Guided system design, requirements, and structured development of the app.
3. **Roger Pressman, Software Engineering: A Practitioner's Approach, 8th Ed., 2014**
4. Helped plan modules and follow proper development methodology.
5. **Karl T. Ulrich, Product Design and Development, 6th Ed., 2016**
6. Supported user-centered design decisions like the Product Details tab.
7. **Material Design Guidelines** – material.io
8. Used for intuitive and consistent bottom navigation design.
9. **UXD World – Bottom Tab Bar Best Practices** – uxdworld.com
10. Guided the choice of a fixed bottom navigation with 3–5 tabs for easy access.
11. **GeeksforGeeks** – [geeksforgeeks.org](https://www.geeksforgeeks.org)
12. Helped implement the navigation bar and Product Details updates.
13. **Stack Overflow** – stackoverflow.com
14. Solved coding challenges during app development.
15. **Design Studio UI/UX – E-Commerce Navigation Best Practices** – designstudiouiux.com
16. Supported better browsing and user convenience in the app.
17. **Think with Google** – business.google.com
18. Helped design product pages for better user engagement and conversions.